MICROCOPY RESOLUTION TEST CHART

NS-1963-A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
SELECTED
NOV 3 0 1987
D

THESIS

DESCRIPTION, ANALYSIS AND SIMULATION
OF A NEW REALIZATION OF DIGITAL FILTERS

by

Ahmed Fahmy Amin Mahrous

September 1987

Dissertation Supervisor: G. A. Myers

87 11 16 019

SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT
DECLASSIFICATION/DOWNGRADING SCHEDULE	Approved for public release, distribution is unlimited

PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)

NAME OF PERFORMING ORGANIZATION	6b OFFICE SYMBOL (If applicable)	7a NAME OF MONITORING ORGANIZATION
Naval Postgraduate School	Code 62	Naval Postgraduate School

ADDRESS (City, State, and ZIP Code)	7b ADDRESS (City, State, and ZIP Code)
Monterey, California 93943- 5000	Monterey, California 93943- 5000

NAME OF FUNDING/SPONSORING ORGANIZATION	8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO	PROJECT NO	TASK NO.	WORK UNIT ACCESSION NO.

TITLE (Include Security Classification)
DESCRIPTION, ANALYSIS AND SIMULATION OF A NEW REALIZATION OF DIGITAL FILTERS

PERSONAL AUTHOR(S)
Ahmed Fahmy Amin Mahrous

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Ph.D. Dissertation	FROM _____ TO _____	September 1987	185

SUPPLEMENTARY NOTATION

COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	IIR, FIR, delta modulation, delta convolution, reduced delta convolution, digital filters

ABSTRACT (Continue on reverse if necessary and identify by block number)

This research considers a new realization of digital filters suitable for VLSI implementation. The method involves delta modulation which provides analog-to-digital (binary) conversion. The output of a linear system is the convolution of the input and the system impulse response. This new digital filter requires that both the input and the impulse response be first converted to bit streams using delta modulation. These bit streams are then convolved. The result is an analog voltage which approximates the convolution of the analog functions.

Direct convolution of the bit streams is difficult to realize with electrical circuits. A greatly simplified system with equivalent performance is a result of this research. This is called the reduced delta convolution (RDC) system (digital filter). The performance of the RDC system when used as a convolver and as a correlator is analyzed and verified by computer simulation. Analyses of the effects of self noise and external noise are included. Conclusions are that the RDC system has a considerable potential as a digital filter when using integrated circuits. Realization requires considerably fewer components and simpler connections than other digital filters. A reason is that there are no multipliers required in the RDC system. The RDC system requires no synchronization, operates in real time and is easily programmed. Further, the RDC system has noise performance which is better than predicted by ordinary filter theory.

Description, Analysis and Simulation
of a New Realization of Digital Filters

by

Ahmed Fahmy Amin Mahrous
Colonel, Egyptian Airforce
B.S., Military Technical College, Cairo, 1970
M.S., Cairo University, 1976

Submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY
IN ELECTRICAL AND COMPUTER ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September 1987

Author: _____
Ahmed Fahmy Amin Mahrous

Approved by:

_____   _____
D. E. Kirk   R. W. Hamming
Professor of Electrical   Professor of Computer Science
and Computer Engineering

_____   _____
F. W. Terman   H. M. Fredricksen
Professor of Electrical   Professor and Chairman, Department
and Computer Engineering   of Mathematics

                         _____
                         G. A. Myers
                         Professor of Electrical
                         and Computer Engineering
                         Dissertation Supervisor

Approved by: _____
John P. Powers, Chairman
Department of Electrical and Computer Engineering

_____
Gordon E. Schacher, Dean of Science and Engineering

2

ABSTRACT

This research considers a new realization of digital filters suitable for VLSI implementation. The method involves delta modulation which provides analog-to-digital (binary) conversion. The output of a linear system is the convolution of the input and the system impulse response. This new digital filter requires that both the input and the impulse response be first converted to bit streams using delta modulation. These bit streams are then convolved. The result is an analog voltage which approximates the convolution of the analog functions.

Direct convolution of the bit streams is difficult to realize with electrical circuits. A greatly simplified system with equivalent performance is a result of this research. This is called the reduced delta convolution (RDC) system (digital filter). The performance of the RDC system when used as a convolver and as a correlator is analyzed and verified by computer simulation. Analyses of the effects of self noise and external noise are included. Conclusions are that the RDC system has considerable potential as a digital filter when using integrated circuits. Realization requires considerably fewer components and simpler connections than other digital filters. A reason is that there are no multipliers required in the RDC system. The RDC system requires no synchronization, operates in real time and is easily programmed. Further, the RDC system has noise performance which is better than predicted by ordinary filter theory.

3

TABLE OF CONTENTS

4

LIST OF TABLES

LIST OF FIGURES

9

10

11

LIST OF SYMBOLS

B :	bandwidth of the signal
$conv(i \cdot T_s)$:	the result of convolution at instant $i \cdot T_s$
$corr(i \cdot T_s)$:	the result of correlation at instant $i \cdot T_s$
C_1 :	a constant equal to sampling period multiplied by the step sizes of input signal and the impulse response
C_{pg} :	constant
f_c :	cutoff frequency
f_i :	instantaneous frequency
f_s :	sampling frequency
F :	the oversampling ratio $= \frac{f_s}{2W}$, where W is the signal bandwidth
g :	gain factor
G :	amplification gain of an amplifier
G_P :	prediction gain
$h_c(t)$:	causal impulse response
$h_{cs}(t)$:	staircase representation of the causal impulse response
$h_{ct}(t)$	staircase representation with constant step size
h_i :	the ith bit of $\hat{h}(t)$
$h_s(t)$:	the stepwise approximation of $h(t)$
$h_v(t)$:	staircase representation with variable step size
$h(t)$:	impulse response
$\hat{h}(t)$:	the delta modulation version of $h(t)$
$h(t) * x(t)$:	convolution of $h(t)$ and $x(t)$
$\hat{h}(\tau)\blacktriangle\hat{x}(jT_s - \tau)$:	the delta convolution (normalized) at time jT_s

13

I_{total} :	total current
$I[i \cdot T_s]$:	increment at instant $i \cdot T_s$
k_1 :	a constant equal to $-C_1 \cdot \sum_{i=0}^{n_1-1} h_i$
k_2 :	a constant equal to C_1
k_3 :	a constant equal to $-k_1$
k_4 :	a constant equal to $-k_2$
L_1 :	the time duration of non-zero values of $h(t)$
n_a :	number of bits for which $< R_h(i) >=< R_x(i) >$
n_d :	number of bits for which $< R_h(i) >\neq< R_x(i) >$
n_1 :	the number of bits of $\hat{h}(t)$
n_2 :	the number of bits of $\hat{x}(t)$
N_o :	average noise power at the output of MF
N_i :	average noise power at the input of MF
p_e :	probability of bit error
q_e :	quantizer error
$r[i \cdot T_s]$:	residual at instant $i \cdot T_s$
r_3, r_4 and r_5 :	resistances
R_h :	the impulse response register
$< R_m(i) >$:	the contents of cell i of register m; i equal to ± 1
R_T :	the ratio of the transistors needed using RDC and DMF methods
R_x :	the input signal register
R_{x1} :	the first n_1 stages of register R_x
$R_{xx}(1)$:	autocorrelation function
s :	slope overload factor
sd_1, sd_2 :	standard deviations

14

S_i :   average input signal power

S_p :   peak signal power

T_d :   time duration of a pulse

T_s :   sampling period

V_1, V_2, V_3, V_{cc} and V_{DD} :   voltages

$x(t)$:   input signal

$x_s(t)$:   the stepwise approximation of $x(t)$

x_i :   the ith bit of $\hat{x}(t)$

x_Δ :   the difference between $x_b(t)$ and $x_s(t)$

$x_b(t)$:   band limited input

x_0' :   the delta modulation slope-following capacity

$\hat{x}(t)$:   the binary output of the delta modulator

$y(t)$:   the output of delta demodulator or delta

                                 convolver (correlator)

$y_{cv_1}(t)$:   output of a conventional matched filter

$\Delta conv(i \cdot T_s)$:   change in the discrete convolution values

$\Delta corr(i \cdot T_s)$:   change in the discrete correlation values

$\Delta 2conv(i \cdot T_s)$:   incremental change in the change of the discrete

                                 convolution values

$\Delta 2corr(i \cdot T_s)$:   incremental change in the change of the discrete

                                 correlation values

Δx :   step size of the input signal

Δh :   step size of the impulse response

ϵ_{conv} :   error in convolution

ϵ_g :   granular distortion

ϵ_{so} :   slope overload distortion

15

ϵ_1 :   a constant for a given impulse response and C_1

ϵ_q^2:   the quantizer performance factor

λ : $n_a - n_d$

ρ_1 :   variance normalized autocorrelation function

σ_g^2 :   granular noise power

$\sigma_{g_a}^2$ granular noise power when the input to the delta

modulator is the signal alone

σ_r^2 :   total quantization noise power

$\sigma_{r_a}^2$:   total quantization noise power when the input

to the DM is the signal alone

σ_{so}^2 :   slope overload quantization noise power

$\sigma_{so_a}^2$:   slope overload quantization noise power when the input

to the delta modulator is the signal alone

σ_Δ^2 :   prediction error variance

$|\frac{dx}{dt}|$:   absolute value of the rate of change of $x(t)$

LIST OF ACRONYMS

ADC:	analog-to-digital converter
BBD:	bucket-brigade device
codec:	coder/decoder
CCD	charge coupled device
dc:	direct current
DFT:	discrete Fourier transform
DM:	delta modulation
DMF:	digital matched filter
DAC:	digital-to-analog converter
FET:	field effect transistors
FIR:	finite impulse response
Hz:	hertz
IC:	integrated circuit
IIR:	infinite impulse response
LPF:	low pass filter
MF:	matched filter
op-amp:	operational amplifier
pdf:	probability density function
PCM:	pulse code modulation
RDC:	reduced delta convolution
RDC LPF:	reduced delta convolution lowpass filter
SNR:	signal to noise ratio
SNRI:	signal to noise ratio improvement
VLSI:	very large scale integrated circuit
XNOR:	exclusive nor

I. INTRODUCTION

This research is concerned with a new realization of digital filters. The method used involves delta modulation.

We define and discuss digital filters in Chapter Two.

In Chapter Three we briefly explain delta modulation which converts an analog signal to a digital (binary) signal. We then introduce a new method of convolution which is called delta convolution in this report. The two signals to be convolved are first converted to binary form using delta modulation. Time domain analysis is used to derive equations of interest and to define the required hardware. Analytical examples are presented to show that the results obtained using the delta convolution method and conventional methods are similar. We conclude that direct implementation of the delta convolution method results in a complicated system (large number of gates and interconnections).

A system which involves two feedback circuits to account for the previous values involved in delta modulation and convolution is developed in Chapter Four. We call this the reduced delta convolution (RDC) system. The result is a simple hardware realization. We define (block diagram) a system which functions as a convolver (linear filter) and a similar one which is a correlator.

In Chapter Five we introduce a possible circuit to implement the RDC system. A comparison between the needed hardware and complexity of the RDC system and a conventional digital matched filter (DMF) is included. The result is that the RDC system has a simple layout for IC realization and requires less than 10 percent of the transistors needed for a typical DMF having comparable performance.

18

Simulation results when the RDC system is used as a convolver (lowpass filter) and a correlator (matched filter) are presented in Chapter Six. The simulation results include the transfer function of a RDC lowpass filter, the response of the RDC lowpass filter to two signals, one in the pass band and the other outside the pass band, and the output of a RDC lowpass filter with a square wave input. Included also are responses of matched filters to a raised cosine pulse and to a chirp signal.

Chapter Seven documents the noise characteristics of the RDC system. Optimum values of important RDC system parameters are derived. Analyses of self noise and input noise are presented. A conclusion made is that the RDC system has noise performance which is better than predicted by ordinary filter theory. Simulation results for the chirp matched filter and lowpass filter when used in the presence of additive input noise are shown. Again, the noise performance is better than predicted by theory. A reason for this is detailed. This chapter also includes a modification to the RDC method called variable step size RDC. Simulation results for the variable step size RDC lowpass filter and notch filter are presented.

The results of this research indicate that the VLSI compatible RDC system may be an attractive means of realizing digital filters. The system requires no synchronization, has no multipliers and is easily programmable. Other features include real time operation (no off-line processing) and no limitations on the time duration of the signal to be represented in matched filter applications.

When compared with other digital filter realizations, the RDC system appears to require considerably fewer components and simpler connections. Finally, because of the properties of delta modulation, the RDC system achieves noise performance which surpasses that predicted by ordinary filter theory.

II. BACKGROUND

This research is concerned with the realization of linear systems (filters) using digital technology. Common realizations of digital filters require a large number of transistors and interconnections. This report describes a new type of digital filter and documents its advantages and performance. We begin this description by introducing linear systems.

A. THEORY OF LINEAR FILTERS

A continuous-time or analog signal may be described by a function of time, say $x(t)$. The response of a linear time-invariant system to $x(t)$ is given by

$$y(t) = \int_{-\infty}^{\infty} h(\tau) \cdot x(t - \tau) \cdot d\tau = \int_{-\infty}^{\infty} x(\lambda) \cdot h(t - \lambda) \cdot d\lambda \qquad (2.1)$$

where $h(t)$ is the response of the system at time t to a unit impulse applied at time $t = 0$. See Fig. 2.1. The function $h(t)$ is, then, used to represent the linear system. Linear systems are used as convolvers to realize filters (lowpass, bandpass, notch) and as correlators (matched filters (MF)).

The right side of Equation 2.1 is commonly called the convolution of two functions $x(t)$ and $h(t)$ and denoted as $x(t) * h(t)$. From transform theory

$$Y(f) = X(f) \cdot H(f) \qquad (2.2)$$

where $Y(f), X(f)$, and $H(f)$ are respectively the Fourier transforms of $y(t), x(t)$, and $h(t)$.

A discrete-time signal is a sequence of numbers $\{x(i \cdot T_s)\}$, where the integer i may vary over a finite or an infinite range, and where T_s is the sampling interval.

A linear time-invariant discrete-time system can be described by the input-output relationship

$$
\begin{aligned}
y(n \cdot T_s) &= T_s \cdot \sum_{m=-\infty}^{\infty} x(m \cdot T_s) \cdot h\left((n-m) \cdot T_s\right) \\
&= T_s \cdot \sum_{k=-\infty}^{\infty} x\left((n-k) \cdot T_s\right) \cdot h(k \cdot T_s)
\end{aligned}
\tag{2.3}
$$

where $x(n \cdot T_s)$ and $y(n \cdot T_s)$ are the input and output signals, and $h(n \cdot T_s)$ is the impulse response of the system. That is, $h(n \cdot T_s)$ is the response of the system at $n \cdot T_s$ due to a unit sample (unit impulse sample) applied at $t = 0$. See Fig. 2.2.

The right side of Equation 2.3 is the convolution sum of the two sequences $\{x(n \cdot T_s)\}$ and $\{h(n \cdot T_s)\}$. When the sequence $\{h(n \cdot T_s)\}$ has only a finite number of non-zero terms, we say the system has a finite impulse response (FIR). Otherwise, the system is said to possess an infinite impulse response (IIR). If $h(n \cdot T_s) = 0$ for $n \leq 0$, we say that the system is causal or physically realizable.

Using z-transforms

$$
Y(z) = H(z) \cdot X(z)
\tag{2:4}
$$

where $Y(z), H(z)$, and $X(z)$ are respectively the z- transforms of $y(n \cdot T_s), h(n \cdot T_s)$ and $x(n \cdot T_s)$.

A large class of linear time-invariant discrete-time systems can also be described by the linear constant coefficient difference equation

$$
y(n \cdot T_s) = \sum_{k=1}^{N} a_k \cdot y\left((n-k) \cdot T_s\right) + \sum_{k=0}^{L} b_k \cdot x\left((n-k) \cdot T_s\right)
\tag{2.5}
$$

where it is possible to convert Equation (2.5) to an equation of the form of Equation (2.3) (Kirk and Strum [1]).

21
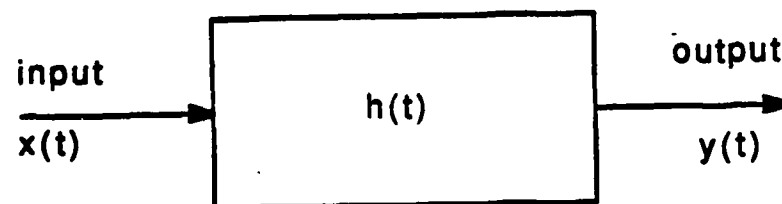
Fig. 2.1. Representation of linear time-invariant continuous-time system.
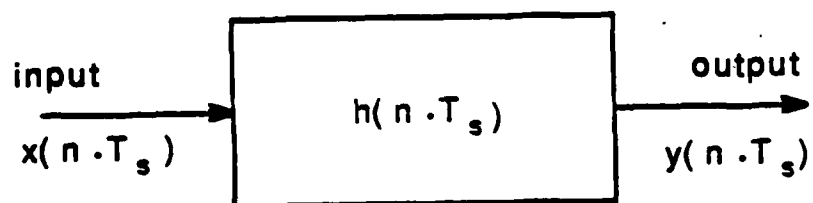
Fig. 2.2. Representation of linear time-invariant discrete-time system.

22

In Equation (2.5), if $a_k = 0$ for all k, then

$$y(n \cdot T_s) = \sum_{k=0}^{L} b_k \cdot x((n-k) \cdot T_s) \qquad (2.6)$$

So, the output of the system depends only upon the current and previous values of the input. This system is called non-recursive.

From these results, we can now consider ways of realizing discrete-time systems. These realizations are called digital filters. This research is concerned with a new realization of a digital filter.

B. VARIOUS REALIZATIONS OF LINEAR FILTERS

The unit sample response for the non-recursive system described in Equation (2.6), is given by

$$h(n \cdot T_s) = \sum_{k=0}^{L} b_k \delta(n-k). \qquad (2.7)$$

This system is known as a finite-impulse response (FIR) system because we have a finite number of terms in the unit sample response (Bird [2]).

From Equation (2.7), a FIR digital filter has a transfer function

$$H(z) = \sum_{k=0}^{L} b_k z^{-k}$$

which is often implemented as a tapped-delay line and is sometimes called a transversal filter. (See Fig. 2.3 and Rabiner [3]).

It is important to realize that, while a non-recursive filter must have a FIR, a FIR filter need not be implemented non-recursively. A recursive implementation of a FIR filter can be obtained by introducing poles in the transfer function and then canceling them with extra zeroes (Bird [2]). A new recursive implementation of a FIR filter, which is called the reduced delta convolution (RDC) method. is introduced in Chapter Four.

23
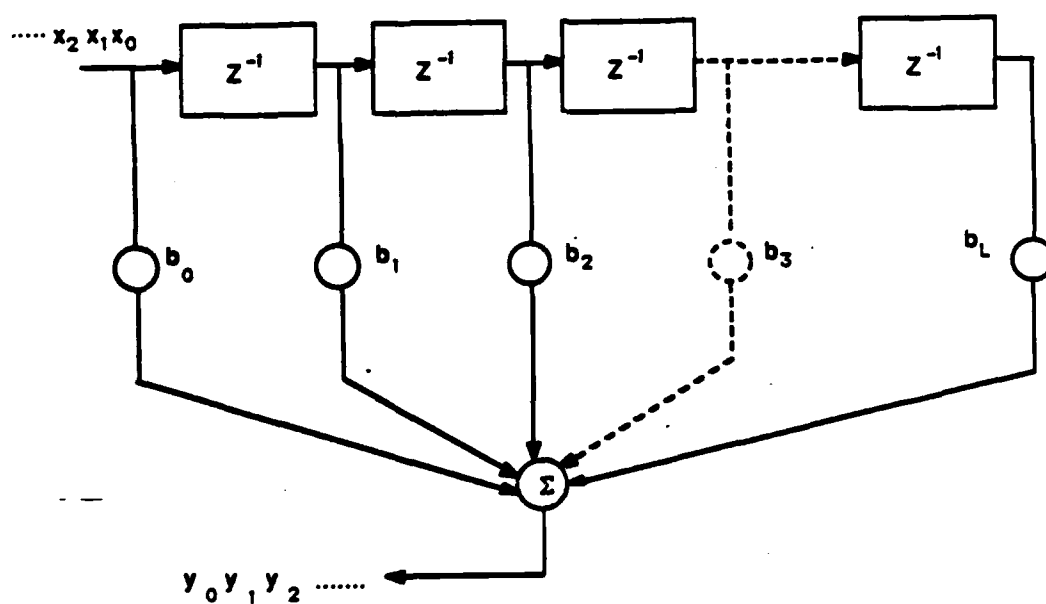
Fig. 2.3. Realization of transversal filter.

A transversal filter can have an arbitrary impulse response of finite time duration and therefore can be used to implement any linear filter. (Any system whose output is linearly related to the input is a linear filter.) In this sense, a transversal filter can be thought of as a fundamental building block of linear systems.

In Equation (2.5), if $a_k \neq 0$ for one value at least, then we have an IIR filter.

In classical signal processing most of the filters are IIR systems. These filters correspond to direct implementation of the filter's rational polynomial transfer function.

In modern signal processing most filters are FIR systems. These filters correspond to a direct implementation of the filter impulse response approximated with a finite number of terms. These filters often arise as matched filters for radar and sonar signals (Kailath [4] and Rabiner [3]).

FIR linear filters have several important properties which make them attractive for digital signal-processing applications. Among these features are simple design, linear phase and the absence of any stability problems which may occur in IIR filters. On the other hand, for the same filter, non-recursive technique uses more taps and multipliers (Hamming [5], Young [6], McClellan [7], and Whalen [8]).

Transversal filters are the popular choice of digital filter when using VLSI. We consider next various circuit realizations of transversal filters.

C. TRANSVERSAL FILTER REALIZATION

1. Block Diagram

a. Transversal Filter as a Linear Filter

To implement a linear filter using transversal filters, it is sufficient to choose the tap weights appropriately. From Equations (2.6) and (2.7), if the desired

impulse response is known, then the transversal filter tap weights are simply the sample values of the filter impulse response. See Fig. 2.4.

b. Transversal Filter as a Matched Filter/Correlator

Pulse detection in noise differs from analog signal transmission in two major respects. First, we are usually concerned with determining the presence or absence of a pulse. Second, we often know the pulse shape in advance, but not its amplitude or arrival time. The pulse-shape information makes it possible to design optimum receiving filters for detecting pulses buried in noise having a known spectral density function. Such optimum filters (correlators) are called matched filters.

Let $x(t), 0 \leq t \leq T_d$, be a signal to which a filter is to be matched. The impulse response of the matched filter is defined as (Turin [9] and Turin [10])

$$
h(t) = \begin{cases} b \cdot x(T_d - t); & 0 \leq t \leq T_d \\ 0; & \text{elsewhere} \end{cases}
$$

where the gain factor b is arbitrary and is henceforth taken as unity and T_d is the pulse time duration. The name matched filter comes from the fact that $h(t)$ has the same shape as the pulse $x(t)$ with time reversed and shifted by T_d seconds. Fig. 2.5 is a diagram showing the application of a matched filter.

The ratio of peak signal power S_p to average noise power N_o at the output of the MF is given by

$$
\frac{S_p}{N_o} = T_d \cdot B \frac{S_i}{N_i}
$$

where S_i and N_i are average input signal power and noise power and B is signal bandwidth. Then

$$
SNRI = \frac{S_p/N_o}{S_i/N_i} = T_d \cdot B \tag{2.8}
$$

where SNRI is the signal-to-noise ratio improvement factor (Horrigan [11]).

26
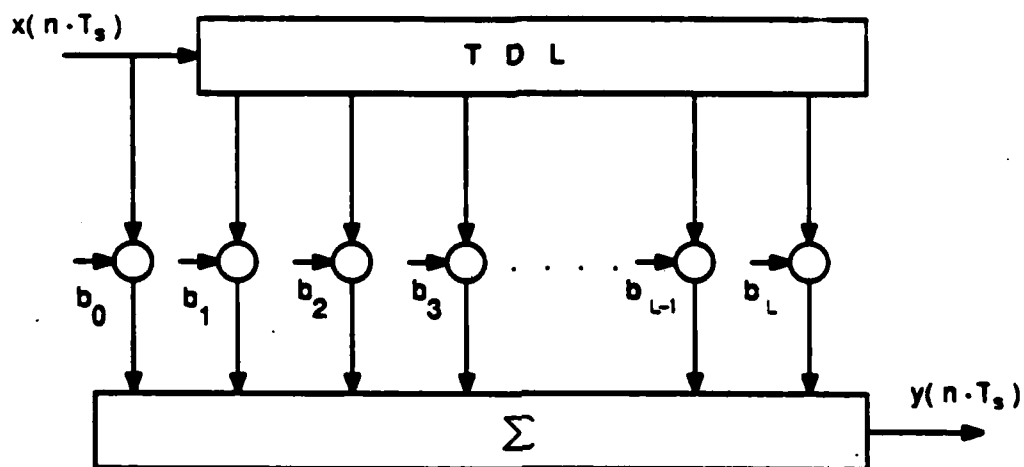
$$b_k = h(k \cdot T_s) ; \quad k = 0,1,2, \ldots ,L$$

The delay line taps are at delay values $= k \cdot T_s$.

Fig. 2.4. Diagram of a transversal filter as a linear filter.

27
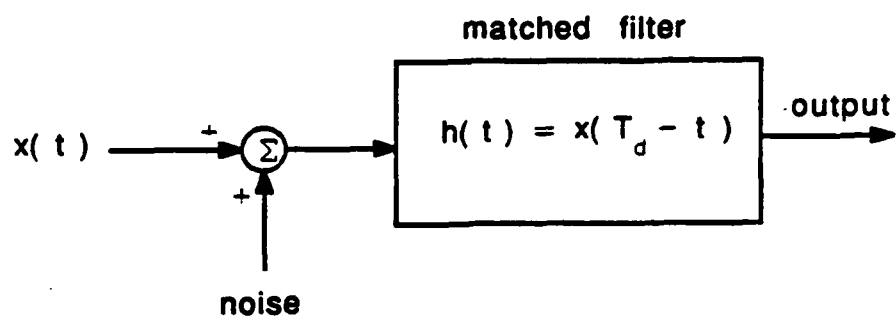
Fig. 2.5. Diagram indicating the application of a matched filter correlator.

This important result shows that the filter SNRI is determined only by the time duration of the pulse and its bandwidth and not by the pulse 'shape'.

Fig. 2.6 shows a transversal filter implementation of a MF, where we can see that the values of the weights are exactly the same as that given for linear filters (Fig. 2.5) except that the positions of the weights are reversed.

2. Circuit Realization

In Figures 2.4 and 2.6 the delay needed for the analog signal can be implemented using the following conventional techniques:

- Charge-Transfer devices (CTD's), which include both charge-coupled devices (CCD's) [12 – 14] and bucket-brigade devices (BBD's) [15, 16], can be used to delay the analog signals (Butler [17]). When CTD's are used to delay analog signals, the signal to be delayed is first sampled at a rate greater than twice the largest significant frequency component of the signal. The analog samples are then clocked through the CTD shift register. CTD's can achieve hundreds of milliseconds of delay (Buss [18]).

- Acoustic delay lines are used as alternatives to CTD's for analog time delay, but for small values of time delays ($< 20\mu$sec (Buss [18])).

- Digital shift registers (SR) preceded by analog/digital conversion (ADC) and followed by digital/analog conversion (DAC) can also be used to delay signals as shown in Fig. 2.7.

a. Transversal Filters Using CTD's

In order to make a CTD transversal filter, it is necessary to nondestructively sample the delay line and to perform the weighted summation. The sampling is achieved in different ways depending on whether CCD's or BBD's are used (Buss [18]).

29
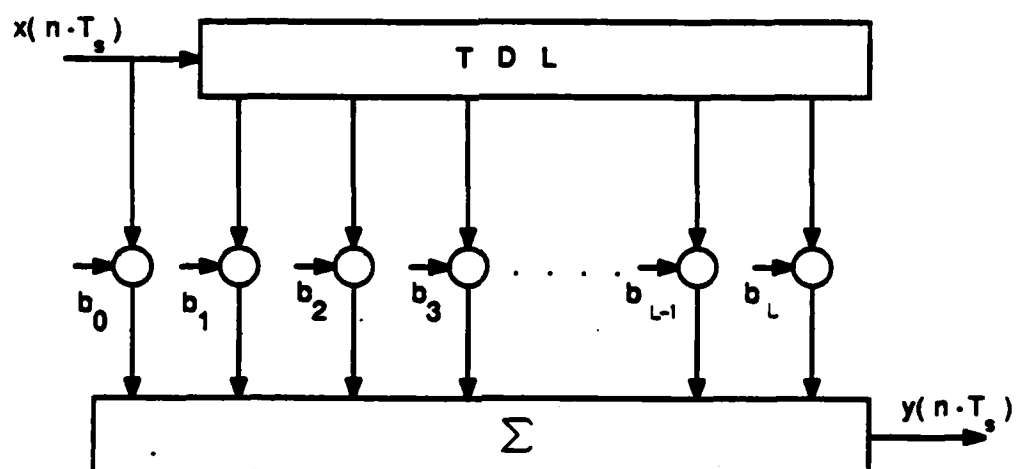
$$b_k = h(T_d - k \cdot T_s) ; \quad k = 0,1,2, \ldots\ldots,L$$

$$T_d = L \cdot T_s$$

The delay line taps are at delay values $= k \cdot T_s$.

Fig. 2.6. Diagram of a transversal filter as a matched filter.

30

Fig. 2.7. Diagram showing realization of time delay of analog signals using digital circuits.

Generally, there are two approaches to implementing programmable transversal filters using CTD's.

The first is the analog/analog system (Bosshart [19], and Denyer [20]). Fig. 2.8 shows a block diagram of an analog/analog CTD transversal filter where analog stores for both input signal and impulse response (reference store) are used in conjunction with analog multipliers.

The main limitation is that since the weighting coefficients are stored in the form of charge, they decay due to thermal leakage and must be refreshed every 10 to 100 ms. The tap weight resolution is limited by the fixed pattern noise associated with the MOS analog multipliers at each tap. In addition, the tap weight data that are stored in off-chip digital memory must be converted to an analog signal before it can be transferred to the on-chip analog reference store.

The other technique is the digital/analog approach (Tiemann [21] and Tanak [22]) where the weighting coefficients are represented in digital form. See Fig. 2.9. This requires a multiplying digital-to-analog converter (MDAC) at every filter point. The MDAC uses MOS transistors that route the output of floating gate taps to either positive or negative summing buses. As a result of this, a fixed pattern noise results due to variations in the tapping transistor characteristics along the length of the array (Gandolfo [23]).

We can summarize the main limitations of transversal filters realized using CTD's as follows:

- The time duration T_d of signals that can be processed using CTD matched filters is ultimately limited by the storage time of the devices (i.e., the time it takes a stored charge to be lost due to leakage). Typically the storage time is of the order of 1 second (Buss [18]).
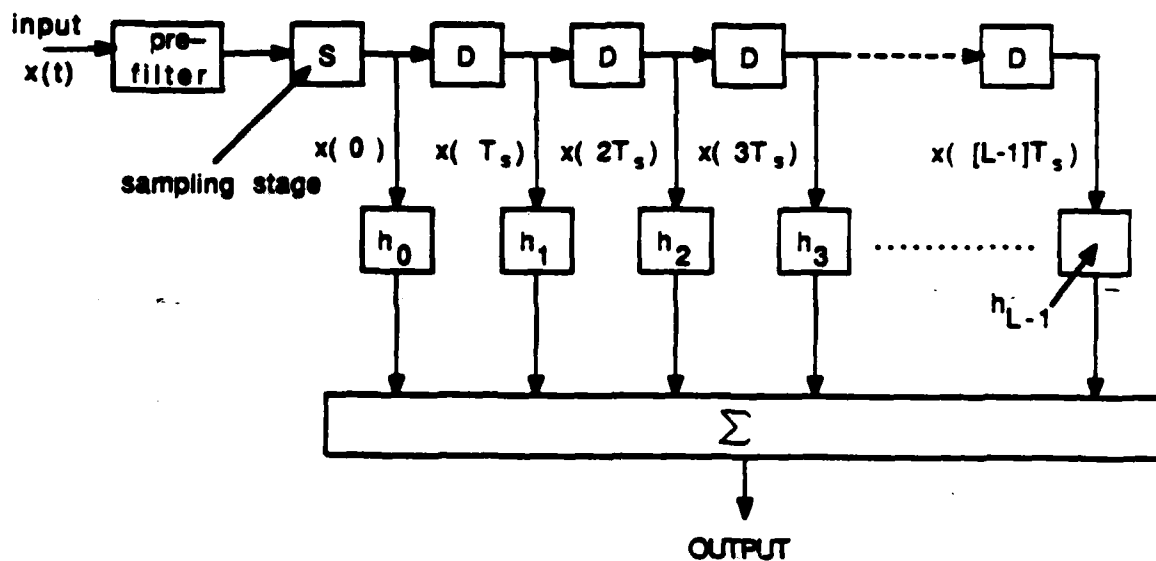
D is a delay stage

(a)
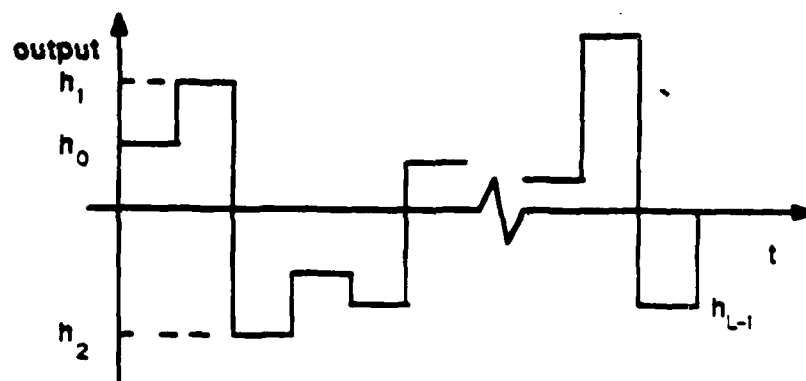
(b)

Fig. 2.8. Diagram of the analog/analog method of implementing CTD transversal filter. (a) System. (b) System impulse response.

33
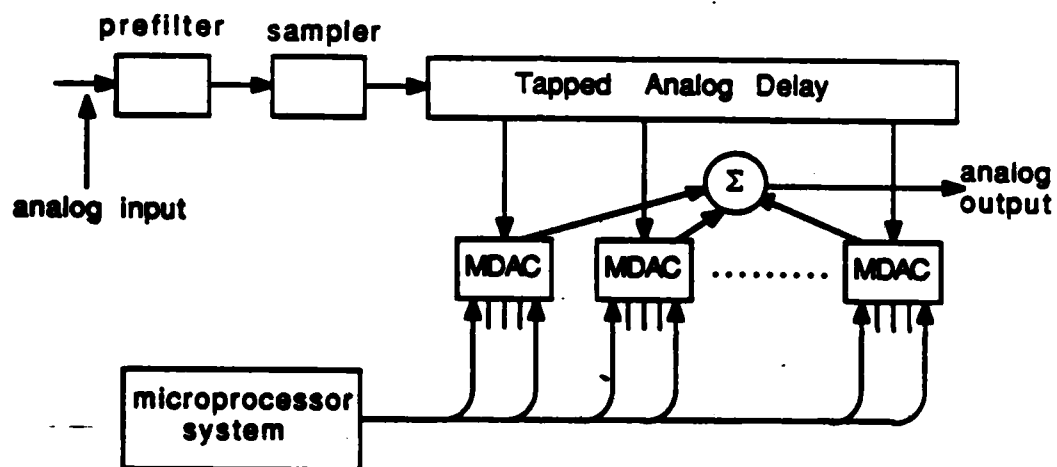
Fig. 2.9. Diagram of the digital/analog method of implementing a CTD transversal filter.

- The filter length (i.e., the number of delay stages L) is ultimately limited by charge transfer inefficiency. Calculations indicate that

$$L \cdot \epsilon < 2 \qquad (2.9)$$

where L is the number of delay stages and ϵ is the charge transfer inefficiency per point where

$$\epsilon = 10^{-3} \text{ for early versions of CCD/MOS}$$

and

$$\epsilon = 0.4 \times (10)^{-3} \text{ for recent versions}$$

(Creasey [24]).

- The Nyquist sampling theorem requires that a signal having a bandwidth B be sampled at a frequency greater than $2B$. Combining this requirement with Equation (2.9) gives the following limitation on the $T_d \cdot B$ product (which is a measure of pulse compression ratio or processing gain) of signals that can be processed using CTD filters.

$$T_d \cdot B < 1/\epsilon. \qquad (2.10)$$

The limitation imposed by Equation (2.10) can be compensated for by selecting the weighting coefficients to invert the dispersion due to the imperfect charge transfer. But the dependence of charge transfer efficiency on the signal amplitude makes it impossible to exactly invert this dispersion at all signal levels.

- The signal bandwidth is limited to less than half the maximum clock frequency of the filter.

  o For CCD's this limitation is $\approx$ 20 MHz.

  o For BBD's this limitation is $\approx$ a few Megahertz.

- Another limitation on the CTD transversal filter is the accuracy with which the weighting coefficients can be determined. For example the transversal filters using MOSFET analog multipliers suffer from:

  o Poor accuracy

  o Lack of long-term stability

  o Drifts that are found in most MOSFET multipliers

The weighting coefficients can also be realized using a split-electrode structure. However, because the weighting coefficients are fixed by the split-electrode structure, such devices are only suitable for applications where a fixed filter response is required (Haken [25]).

b. Transversal Filters Using Digital Delay

It is possible to use clocked shift registers to provide delay. Then in Fig. 2.10 the sampled input $x(n \cdot T_s)$ is quantized into a number of bits, typically 12 to 16 bits for audio and 6 to 8 bits for video signals.

The digital signals (code words) recirculate in virtual delay lines composed of random access memory (RAM) and are multiplied by the weights which are stored in programmable read only memory (PROM). Every sampling time period a new code word enters the RAM where the oldest one is discarded. Using a single multiplier, the multiplication rate has to be $L \cdot f_s$, where L is the filter length and f_s is the sampling frequency. The large number of multiplications required each second is the main disadvantage of this scheme (Kailath [4]).
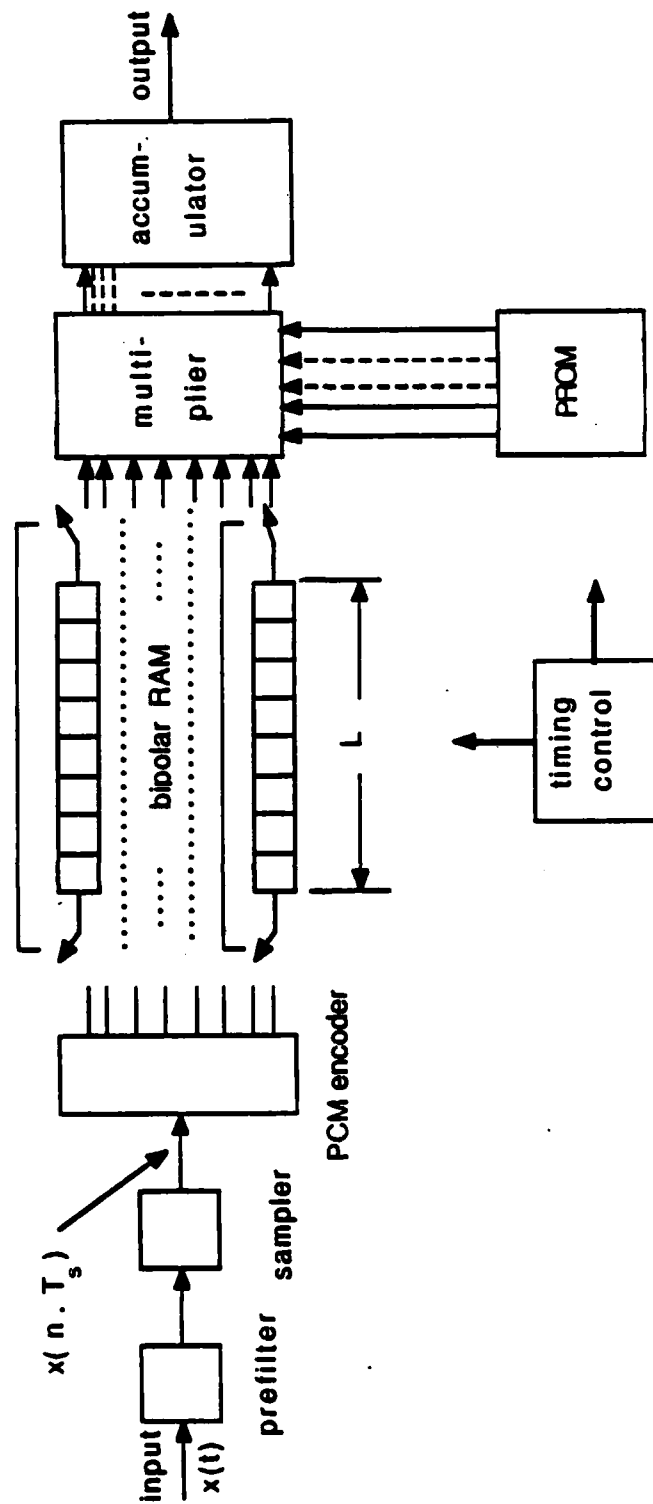
Fig. 2.10. Diagram of a digital transversal filter which uses a single multiplier.

Fig. 2.11 shows an equivalent scheme which uses L multipliers and where m_1 and m_2 are the code word lengths of the PCM encoder and microprocessor system respectively. In this scheme the multiplication rate is reduced to f_s. The main disadvantage of this scheme is the increased cost compared with an equivalent CTD implementation (Terrell [26]).

The next chapter introduces a new approach to the realization of transversal filters. The circuit that results requires no voltage multipliers which results in a considerably reduced number of components (transistors) when compared with CTD and digital delay versions.
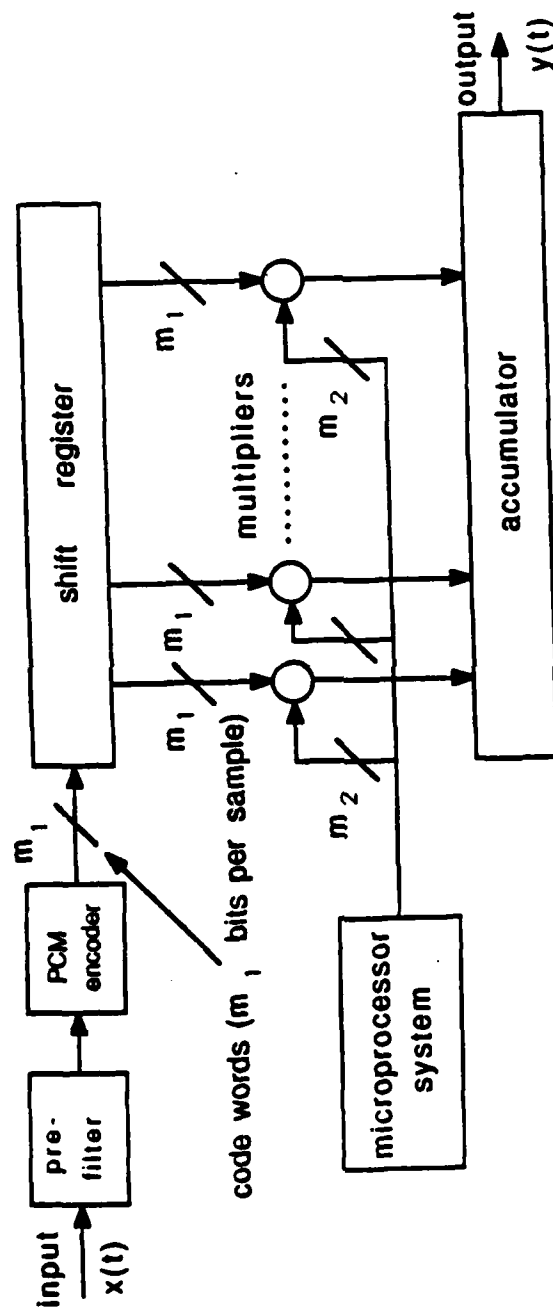
Fig. 2.11. Diagram of a digital transversal filter which uses L multipliers.

III. DELTA CONVOLUTION

A. INTRODUCTION

This chapter introduces a new method of convolution which is called DELTA CONVOLUTION in this report. The two signals to be convolved are delta modulated.

B. DELTA MODULATION

Delta modulation (DM) represents an analog waveform with a binary sequence (bit stream). This analog-to-digital conversion (ADC) is quite different from pulse code modulation (PCM). PCM involves code words and requires somewhat complex encoders and decoders. Recovering the analog waveform from the code words requires synchronization. By contrast, DM involves no code words, requires no timing signals, and uses simple hardware (Roden [27]).

It has been found that analog signals such as speech and video signals generally have a considerable amount of redundancy; that is, there is a significant correlation between successive samples when these signals are sampled at a rate higher than the Nyquist rate. The redundancy in these analog signals makes it possible to predict a sample value from preceding sample values and to transmit the difference between the actual sample value and the predicted sample value estimated from the past samples. This results in a technique called DIFFERENCE ENCODING. DM is one of the simplest forms of difference encoding (Ha [28]).

The idealized DM codec (coder/decoder) is shown in Fig. 3.1. The band limited output of the LPF is compared by subtraction with the stepwise approximation $x_s(t)$. The difference x_Δ is passed through a hard limiter whose output is

$$\begin{pmatrix} +1 \\ -1 \end{pmatrix} \text{ if } \begin{pmatrix} x_b(t) > x_s(t) \\ x_b(t) < x_s(t) \end{pmatrix}.$$

The resulting binary output $\hat{x}(t)$ is applied through an ideal integrator (or accumulator) with a feedback gain factor G to produce $x_s(t)$ where

$$G = \begin{cases} \Delta x & \text{when using an accumulator} \\ \frac{\Delta x}{T_s} & \text{when using an integrator; } T_s \text{ is the sampling interval} \end{cases} \tag{3.1}$$

and where

$$x_s\left(\frac{2n+1}{2}T_s\right) = \begin{cases} \Delta x \cdot \sum_{i=0}^{n} \hat{x}\left(\frac{2i+1}{2}T_s\right); & \hat{x}\left(\frac{2i+1}{2}T_s\right) \in \{-1, +1\} \\ & \text{when using an accumulator} \\ \frac{\Delta x}{T_s} \cdot \int_{-\infty}^{nT_s} \hat{x}(t) \cdot dt & \text{when using an integrator} \end{cases} \tag{3.2}$$

where

$$x_s(t) = x_s\left(\frac{2n+1}{2}T_s\right); n \cdot T_s < t < (n+1) \cdot T_s \qquad \forall n$$

The decoder for a DM waveform is simply a staircase generator (accumulator) or an integrator. If a "1" is received the staircase increments positively. If a "0" is received the staircase increments negatively. The accumulator (or integrator) is followed by an amplifier having an amplification factor G. The amplifier may be followed by a low pass filter (LPF) to smooth the staircase output into a continuous function. A set of waveforms associated with the DM are shown in Fig. 3.2 which also defines Δx of Equation (3.1).

DM systems are subject to two types of quantization distortion which are generally referred to as granular quantization noise and slope overload noise. Since

41
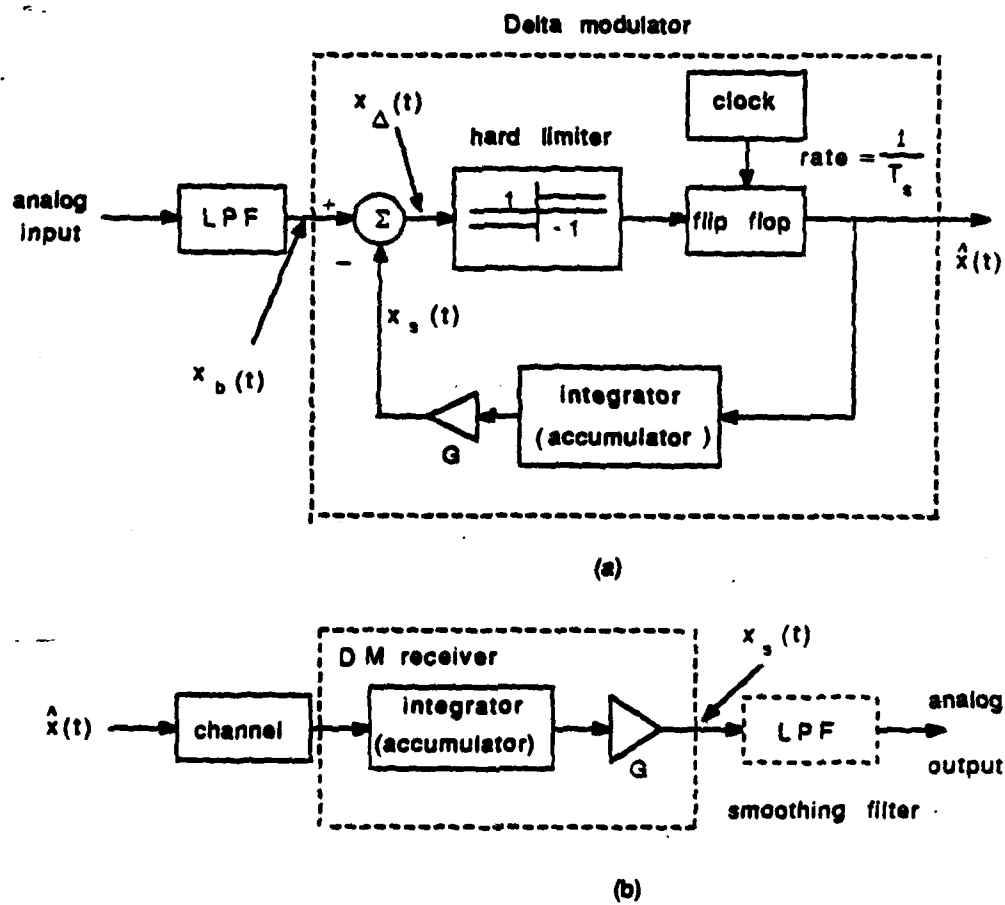
Fig. 3.1. Block diagram of the delta modulation codec.
(a) Encoder. (b) Decoder.

42
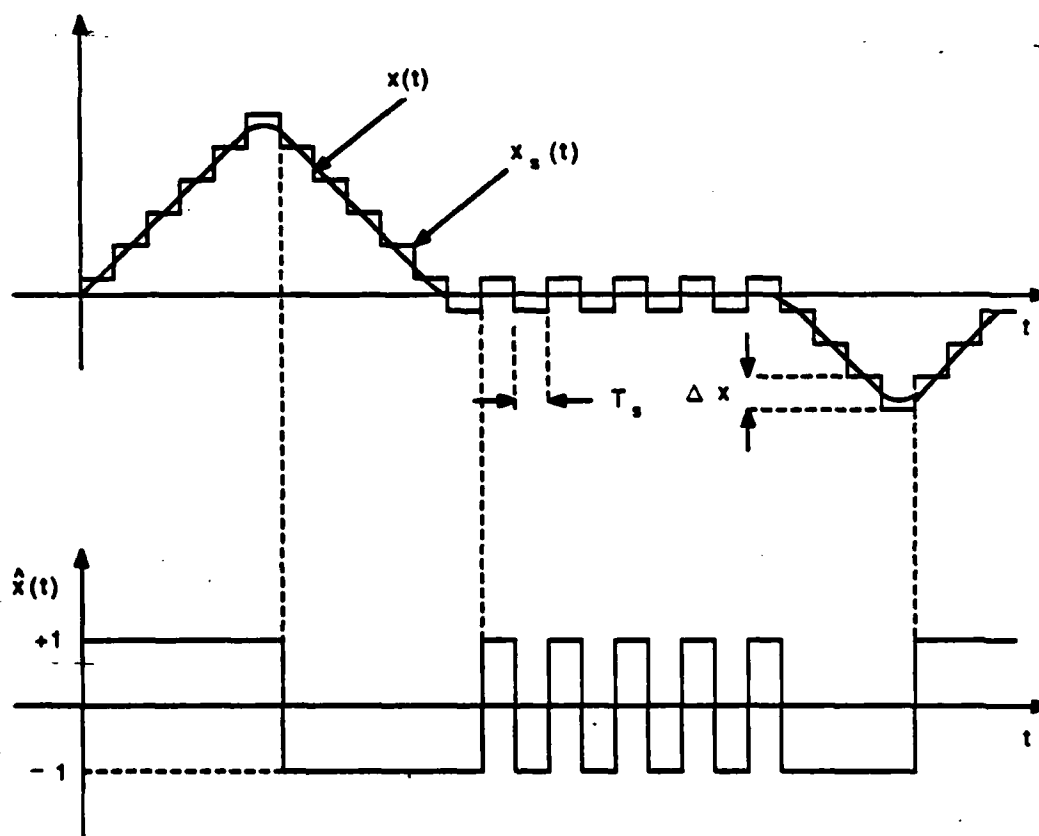
Fig. 3.2. Delta modulation waveforms.

43

$x_s(t)$ cannot change by more than Δx units in T_s seconds, $\frac{\Delta x}{T_s}$ is the highest input signal rate-of-change that the DM codec can follow. We call $\frac{\Delta x}{T_s}$ the DM slope-following capacity and denote it by x'_0. When $|\frac{dx}{dt}|$ exceeds this quantity, slope overload occurs and gives rise to the kind of error shown in Fig. 3.3. The granular noise arises because the DM signal is a discrete-amplitude representation of a continuous amplitude process (Goodman [29] and Greenstein [30]). Consequently, granular noise is always present. Only granular noise occurs when the input is changing very slowly ($\approx$ constant).

The key to effective use of delta modulation is the intelligent choice of two parameters: (1) the step size Δx, and (2) the sampling interval T_s. Obviously, granular noise is reduced by decreasing Δx, but at the expense of a reduced slope-following capacity, and, hence, greater slope overload noise. In Chapter 7 we introduce a procedure to choose the optimum step size and sampling interval such that the total noise obtained from the granular noise and the slope overload noise is minimized.

C. DELTA CONVOLUTION

The subject of this research is the representation of any linear filter using DM. Time domain analysis is used to derive equations of interest and to define the required hardware. In this chapter we pursue the DM equivalent of a linear filter by direct operation on the convolution integral. This results in a cumbersome expression and complicated hardware. However, working from this first result, it is possible to apply two successive modifications which result in a quite simple equation and a simple hardware structure. These modifications are presented in Chapter 4.
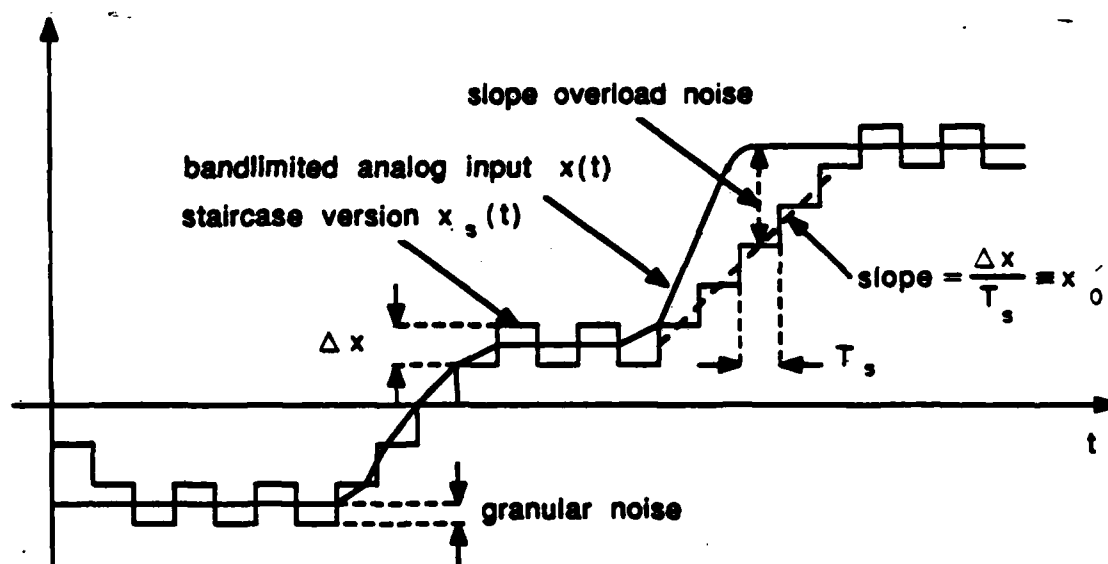
Fig. 3.3. Waveforms showing slope overload and granular noise.

Assume that $h(t)$ and $x(t)$ are two analog signals where

$$h(t) = x(t) = 0; \qquad t \leq 0$$

Let $\hat{h}(t)$ and $\hat{x}(t)$ be the delta modulated versions of $h(t)$ and $x(t)$, where $h_s(t)$ and $x_s(t)$ are the staircase representations of $h(t)$ and $x(t)$ such that we have no slope overload (Fig. 3.2). Assume also that

$$h_i = i^{\underline{th}} \text{ bit of } \hat{h}(t) = \hat{h}(t) \text{ for } i \cdot T_s < t < (i+1)T_s \qquad (3.3)$$

and

$$x_i = i^{\underline{th}} \text{ bit of } \hat{x}(t) = \hat{x}(t) \text{ for } i \cdot T_s < t < (i+1) \cdot T_s \qquad (3.4)$$

We let h_i and $x_i \in \{-1, +1\}$ for all i. Therefore,

$$h_s(t) = \Delta h \cdot \sum_{i=0}^{j} h_i; \qquad j \cdot T_s < t < (j+1) \cdot T_s \qquad (3.5)$$

and

$$x_s(t) = \Delta x \cdot \sum_{i=0}^{j} x_i; \qquad j \cdot T_s < t < (j+1) \cdot T_s \qquad (3.6)$$

When the sampling rate and step sizes are chosen properly, then, as seen in Fig. 3.2,

$$x(t) \approx x_s(t); \qquad j \cdot T_s < t < (j+1) \cdot T_s. \qquad (3.7)$$

Therefore, from Equations (3.6) and (3.7) we will have

$$x(t) \approx x_s(t) = \Delta x \cdot \sum_{i=0}^{j} x_i; \qquad j \cdot T_s < t < (j+1) \cdot T_s \qquad (3.8)$$

where

$$j = \text{integer part of } \left[\frac{t}{T_s}\right].$$

In the same manner,

$$h(t) \approx h_s(t) = \Delta h \cdot \sum_{i=0}^{j} h_i; \qquad j \cdot T_s < t < (j+1) \cdot T_s \qquad (3.9)$$

46

We now have expressions for $x(t)$ and $h(t)$ which involve the bits of $\hat{x}(t)$ and $\hat{h}(t)$.

Now, a linear system (filter) is characterized by its impulse response $h(t)$. Given an input voltage $x(t)$ and impulse response $h(t)$, the output voltage $y(t)$ can be written as the convolution of $x(t)$ and $h(t)$. That is,

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau) \cdot x(t - \tau) \cdot d\tau$$

A conclusion is that in the time domain, linear filters are completely and uniquely identified by their impulse response $h(t)$.

The preceding development suggests that $x(t) \approx x_s(t)$ and $h(t) \approx h_s(t)$ and so we can write

$$y(t) \approx h_s(t) * x_s(t) = \int_{-\infty}^{\infty} h_s(\tau) \cdot x_s(t - \tau) \cdot d\tau \tag{3.10}$$

Let $k = $ integer part of $\left[\frac{\tau}{T_s}\right]$. Then from Equation (3.9)

$$h_s(\tau) = \Delta h \cdot \sum_{p=0}^{k} h_p; \qquad k \cdot T_s < \tau < (k+1) \cdot T_s \tag{3.11}$$

Similarly, from Equation (3.8)

$$x_s(t - \tau) = \Delta x \cdot \sum_{i=0}^{m} x_i; \qquad m \cdot T_s < t - \tau < (m+1)T_s \tag{3.12}$$

where $m = $ integer part of $\left[\frac{t-\tau}{T_s}\right]$. But $j = $ integer part of $\left[\frac{t}{T_s}\right]$ and so

$$t = j \cdot T_s + \in_1; \qquad \in_1 \text{ is positive and less than } T_s. \tag{3.13}$$

But $k = $ integer part of $\left[\frac{\tau}{T_s}\right]$ and so

$$\tau = k \cdot T_s + \in_2; \qquad \in_2 \text{ is positive and less than } T_s. \tag{3.14}$$

47

From Equations (3.13) and (3.14)

$$\text{integer part of } \left[\frac{t-\tau}{T_s}\right] = \text{integer part of } \left[\frac{(j-k)\cdot T_s + (\epsilon_1 - \epsilon_2)}{T_s}\right]$$

because ϵ_1 and ϵ_2 are positive and less than T_s. So,

$$\text{the integer part of } \left[\frac{t-\tau}{T_s}\right] = j - k \qquad (3.15)$$

From Equations (3.12) and (3.15)

$$x_s(t-\tau) = \Delta x \cdot \sum_{i=0}^{j-k} x_i; \qquad (j-k)\cdot T_s < t - \tau < (j-k+1)\cdot T_s \qquad (3.16)$$

From Equations (3.10), (3.11) and (3.16)

$$y(t) \approx \int_{-\infty}^{\infty} \left[\Delta h \cdot \sum_{p=0}^{k} h_p\right] \cdot \left[\Delta x \cdot \sum_{i=0}^{j-k} x_i\right] \cdot d\tau$$

$$\approx \int_{0}^{L_1} \left[\Delta h \cdot \sum_{p=0}^{k} h_p\right] \cdot \left[\Delta x \cdot \sum_{i=0}^{j-k} x_i\right] \cdot d\tau \qquad (3.17)$$

where L_1 is the time duration of non-zero values of $h(t)$. A duration of L_1 seconds corresponds to $n_1 = \frac{L_1}{T_s}$ bits of $\hat{h}(t)$.

A discrete version of $y(t)$ is formed by replacing the integral in Equation (3.17) with another summation to obtain

$$y(t) \approx y([j+1]\cdot T_s) = \Delta x \cdot \Delta h \cdot T_s \cdot \sum_{k=0}^{n_1-1} \left\{\left[\sum_{p=0}^{k} h_p\right] \cdot \left[\sum_{i=0}^{j-k} x_i\right]\right\}; \qquad j = 0,1,2,\ldots$$

$$y(t) \approx y(j \cdot T_s) = \Delta x \cdot \Delta h \cdot T_s \cdot \sum_{k=0}^{n_1-1} \left\{\left[\sum_{p=0}^{k} h_p\right] \cdot \left[\sum_{i=0}^{j-k-1} x_i\right]\right\}; \qquad j = 1,2,3,\ldots$$

$$= 0 \quad \text{for } j \leq 0 \qquad (3.18)$$

We write $y(t) \approx y(j \cdot T_s) = C_1 \cdot \left\{\hat{h}(\tau)\triangle\hat{x}(j \cdot T_s - \tau)\right\}; \qquad j = 1,2,3,\ldots$

$$= 0 \qquad\qquad j \leq 0 \qquad (3.19)$$

48

where

$$C_1 = \Delta x \cdot \Delta h \cdot T_s$$

and

$$\hat{h}(\tau)\Delta\hat{x}(j \cdot T_s - \tau) = \sum_{k=0}^{n_1-1} \left\{ \left[\sum_{p=0}^{k} h_p \right] \cdot \left[\sum_{p=0}^{j-k-1} x_p \right] \right\}$$

where h_p and $x_p \in \{-1, +1\}$.

We call $\hat{h}(\tau)\Delta\hat{x}(j \cdot T_s - \tau)$ the delta convolution (normalized) at time $j \cdot T_s$. In this last form, $y(t)$ can be represented as the sum of appropriate products of ± 1. This result provided the initial motivation for this research because the product $(\pm 1) \cdot (\pm 1)$ can be formed with XNOR gates. This is a considerable hardware advantage compared with usual digital multipliers (and usual digital filters) which have to form the product of PCM code words.

As an example of delta convolution, we convolve $h(t)$ with $x(t)$ where $h(t)$ and $x(t)$ are given in Fig. 3.4. Applying the conventional methods of convolution the result $y_{cv_1}(t)$ is given in Fig. 3.5 when we let $T_s = 1$.

Assume that $\Delta x = \Delta h = 1$. Then, the delta modulation versions of $h(t)$ and $x(t)$ are given in Fig. 3.6. Applying the delta convolution Equation (3.18),

$$y(0) = 0.$$

The delta convolution when $j = 1$ (Fig. 3.7) is given by Equation (3.18) as

$$y(T_s) = C_1 \cdot \left\{ \hat{h}(\tau)\Delta\hat{x}(T_s - \tau) \right\}$$

$$= C_1 \cdot \{h_0 \cdot x_0\}$$

$$= C_1 \cdot \{1 \cdot 1\}$$

$$= C_1 = 1 \text{ with our assumptions.}$$

49
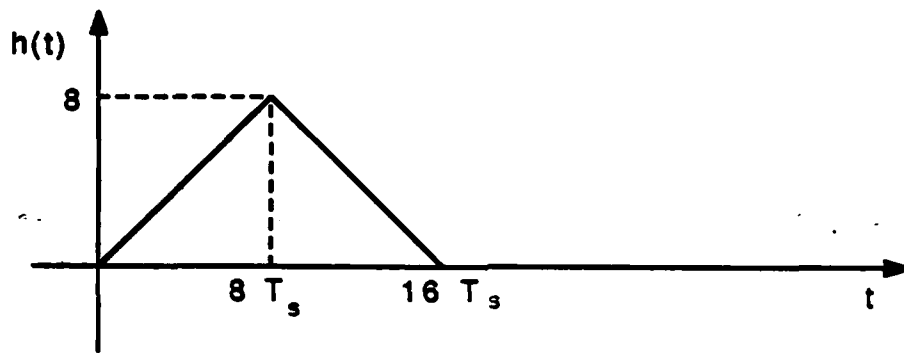
(a)
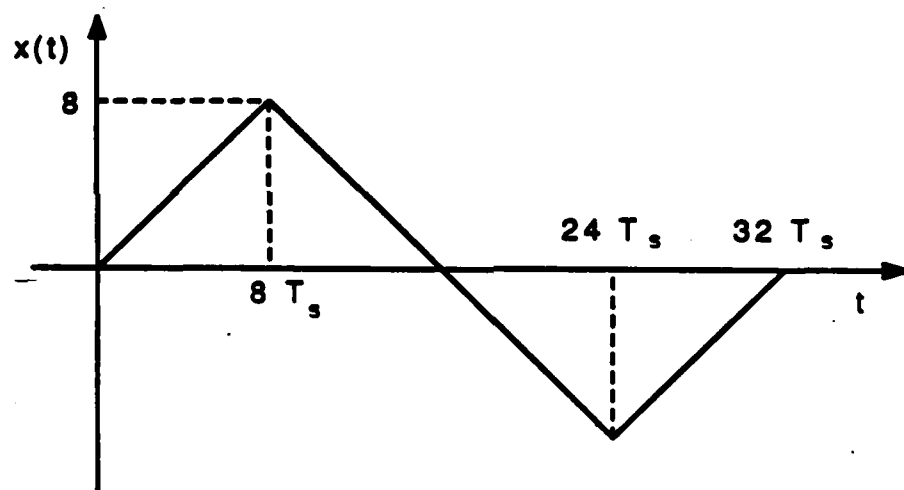
(b)

Fig. 3.4. Waveforms. (a) Impulse response. (b) Input signal.

50
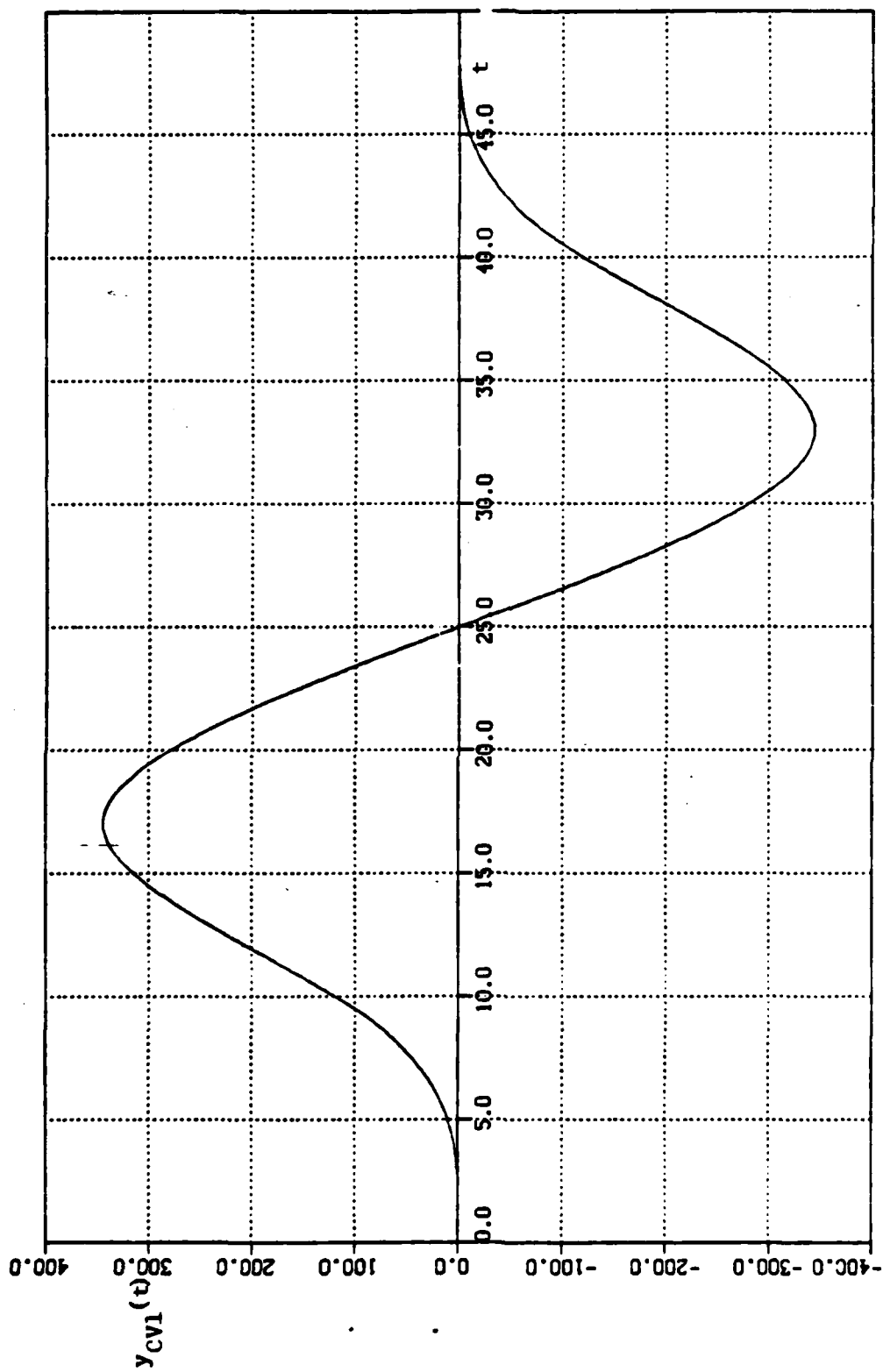
Fig. 3.5. The result of convolution using the conventional methods.

51
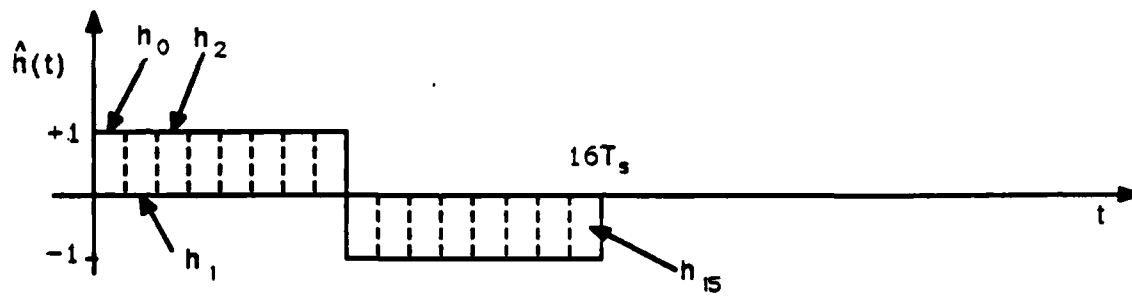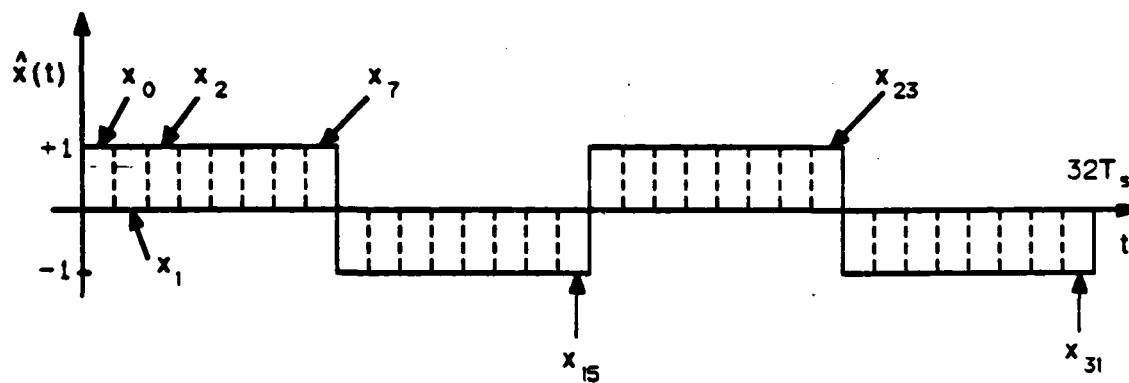
Fig. 3.6. Delta modulation versions. (a) Impulse response. (b) Input signal.
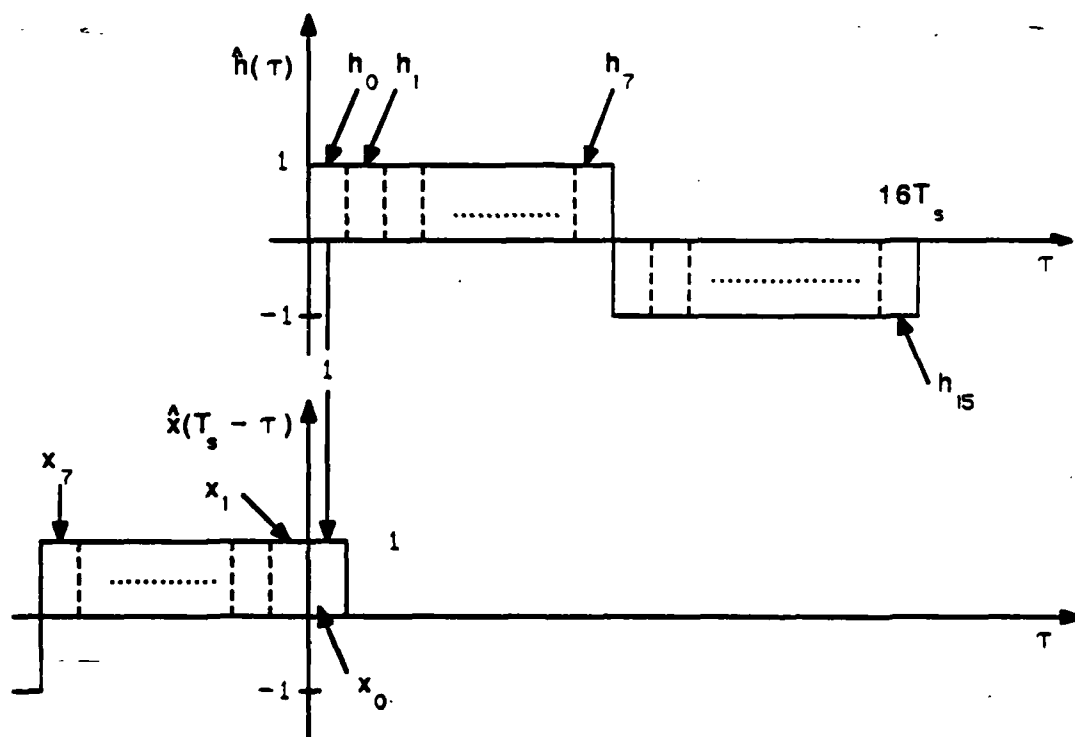
Fig. 3.7. The needed products for delta convolution at time T_s.

The delta convolution when $j = 2$ (Fig. 3.8) is given by Equation (3.18) as,

$$y(2T_s) = C_1 \cdot \left\{ \hat{h}(\tau) \triangle \hat{x}(2T_s - \tau) \right\}$$

$$= C_1 \cdot \{ h_0(2x_0 + x_1) + h_1 x_0 \}$$

$$= C_1 \cdot \{ 1(2 + 1) + 1(1) \}$$

$$= C_1 \cdot (4) = 4.$$

The delta convolution when $j = 3$ (Fig. 3.9) is given by Equation (3.18) as

$$y(3T_s) = C_1 \cdot \left\{ \hat{h}(\tau) \triangle \hat{x}(3T_s - \tau) \right\}$$

$$= C_1 \cdot \{ h_0(3x_0 + 2x_1 + x_2) + h_1(2x_0 + x_1) + h_2 x_0 \}$$

$$= C_1 \cdot \{ 1(3 + 2 + 1) + 1(2 + 1) + 1(1) \}$$

$$= C_1 \cdot 10 = 10.$$

In the same way we can find the result of delta convolution $y(t)$ for all values of $j \cdot T_s$ (Fig. 3.10). It is clear from Figures 3.5 and 3.10 that simulated results of analog convolution and delta convolution are essentially the same for this example. Note that the maximum value of the convolution is 344.0 for both simulations. Theoretically, the maximum value of the convolution is 341.33.

It is possible to expand Equation (3.18) and then create a circuit which forms $y(j \cdot T_s)$ for any $\hat{h}(t)$ and $\hat{x}(t)$. This is done in Appendix A. The result is a circuit having an unacceptably large number of XNOR gates and a large memory. We present in Chapter 4 a much simplified form of Equation (3.18) which eliminates the considerable redundancy of mathematical operations inherent in that equation. The practical result is a simple hardware realization of a circuit to form $y(j \cdot T_s)$.
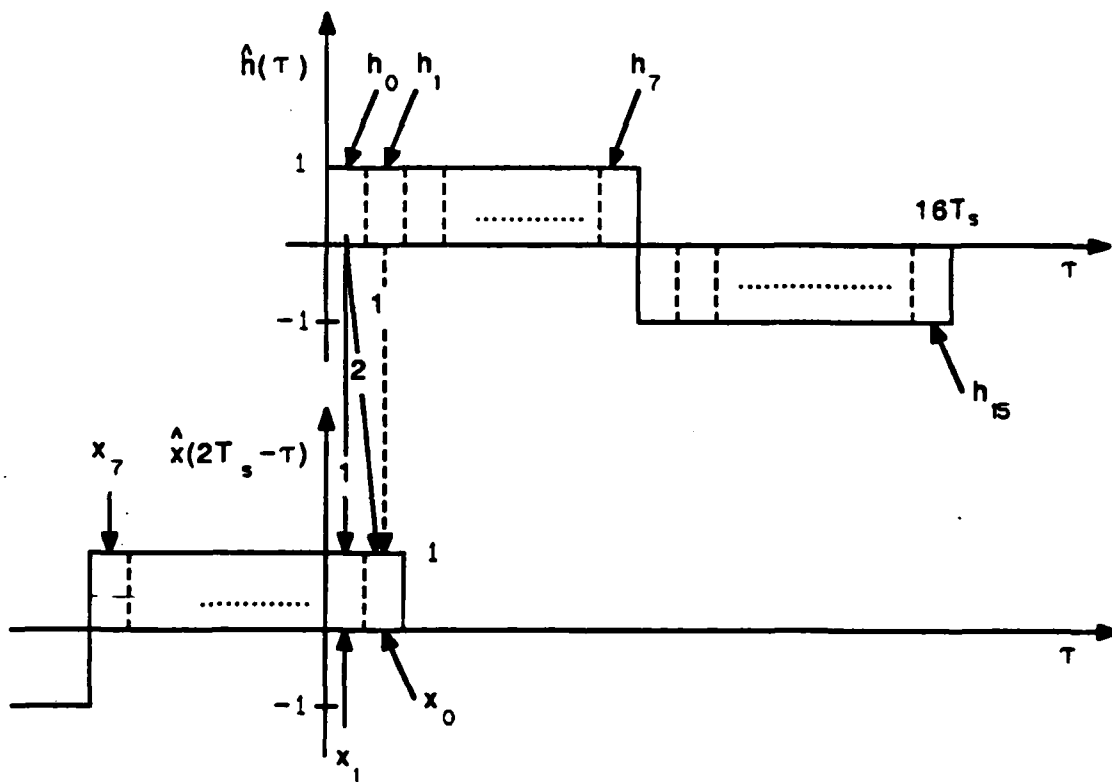
Fig. 3.8. The needed products for delta convolution at time $2T_s$.
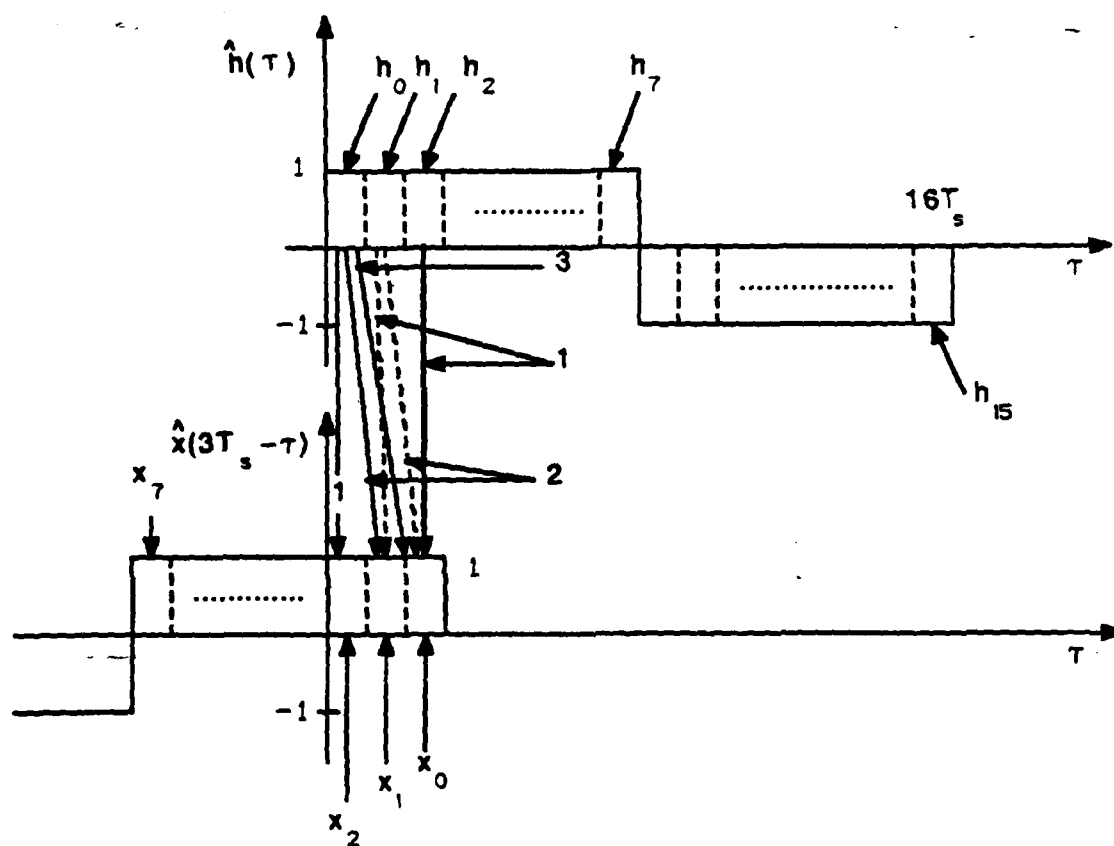
Fig. 3.9. The needed products for delta convolution at time $3T_s$.
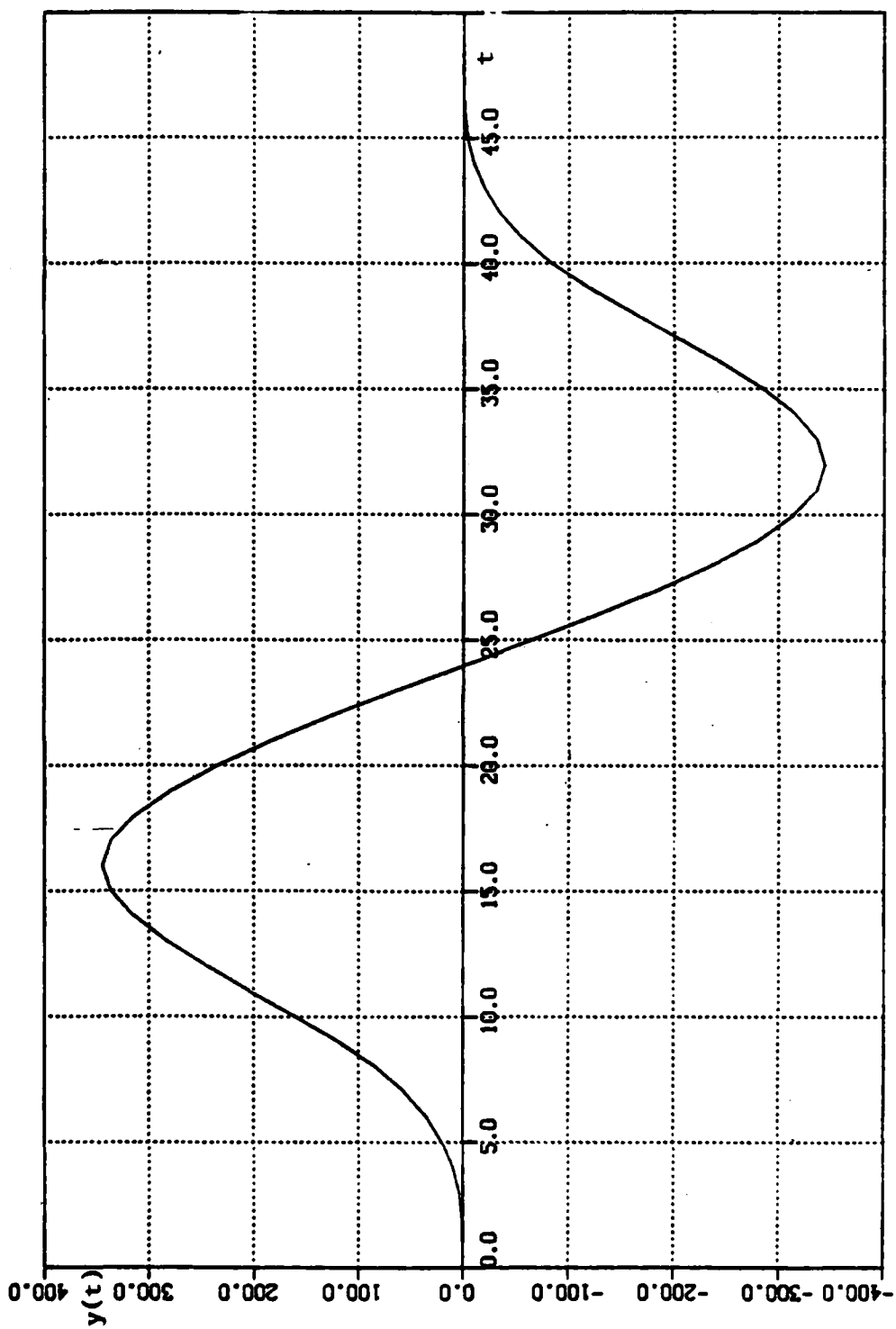
Fig. 3.10. The result of convolution using the delta convolution method.

IV. REDUCED DELTA CONVOLUTION

Implementing Equation (3.18) directly results in a large number of gates and interconnections as noted in Appendix A. The following reasoning suggests a simpler form may be possible. The delta modulation representation of a function ($x(t)$ or $h(t)$) at any time t involves all previous values (history) of the function. The feedback portion of the delta modulator accounts for this history. Also, the result at any time t of convolving two functions depends on the history of both functions. The integral portion of the convolution equation accounts for this history.

When using Equation (3.18) to determine the delta convolution result at time $t = t_2, t_3$, etc., we essentially involve (calculate) repeatedly the previous history of both $x(t)$ and $h(t)$ at t_2, t_3, etc. This implies considerable redundancy of calculation (hardware) which increases with time. By carrying the history (previous calculations) forward to the next calculation, the required hardware should be reduced. Feedback can be used for this purpose. Indeed, use of one feedback circuit does reduce the complexity as shown in Appendix B. In this chapter, we develop from Equation (3.18) a system which involves two feedback circuits to account for the previous values involved in delta modulation and convolution. We call this the reduced delta convolution (RDC) system.

The RDC system is a particularly simple implementation of Equation (3.18) involving only n_1 gates and two registers the lengths of which are n_1 and $n_1 + 1$ stages where n_1 is the number of bits used to represent the system impulse response $h(t)$. The form of the RDC system can be developed by considering the terms of Equation (3.18) at times $T_s, 2T_s$ and $3T_s$ as follows:

$$y(T_s) = C_1 \cdot [x_0 \cdot h_0] \triangleq r[T_s]$$

$$y(2T_s) = C_1 \cdot [h_0 \cdot (2x_0 + x_1) + h_1 \cdot x_0]$$

$$= 2r[T_s] + r[2T_s]$$

$$= y(T_s) + I[2T_s]$$

where

$$r[2T_s] = C_1 \cdot [h_0 \cdot x_1 + h_1 \cdot x_0]$$

$$I[2T_s] = r[T_s] + r[2T_s]$$

$$= I[T_s] + r[2T_s]$$

$$I[T_s] = r[T_s]$$

and

$$y(3T_s) = C_1 \cdot [h_0 \cdot (3x_0 + 2x_1 + x_2) + h_1 \cdot (2x_0 + x_1) + h_2 \cdot x_0]$$

$$= 3r[T_s] + 2r[2T_s] + r[3T_s]$$

$$= y(2T_s) + I[3T_s]$$

where

$$r[3T_s] = C_1 \cdot [h_0 \cdot x_2 + h_1 \cdot x_1 + h_2 \cdot x_0]$$

$$I[3T_s] = r[T_s] + r[2T_s] + r[3T_s]$$

$$= I[2T_s] + r[3T_s]$$

Note that the present value of the output is the previous value plus an increment I. Further, the present value of I is the previous value plus a residual r which depends on the present contents of the input register. So, the new data updates r which then updates I which then updates the output. These updates are accomplished with feedback which involves storage of a discrete voltage as shown in Fig. 4.1 where $y(j \cdot T_s) \approx y(t)$, and clock rate $= \frac{1}{T_s}$. Although Fig. 4.1 has

59
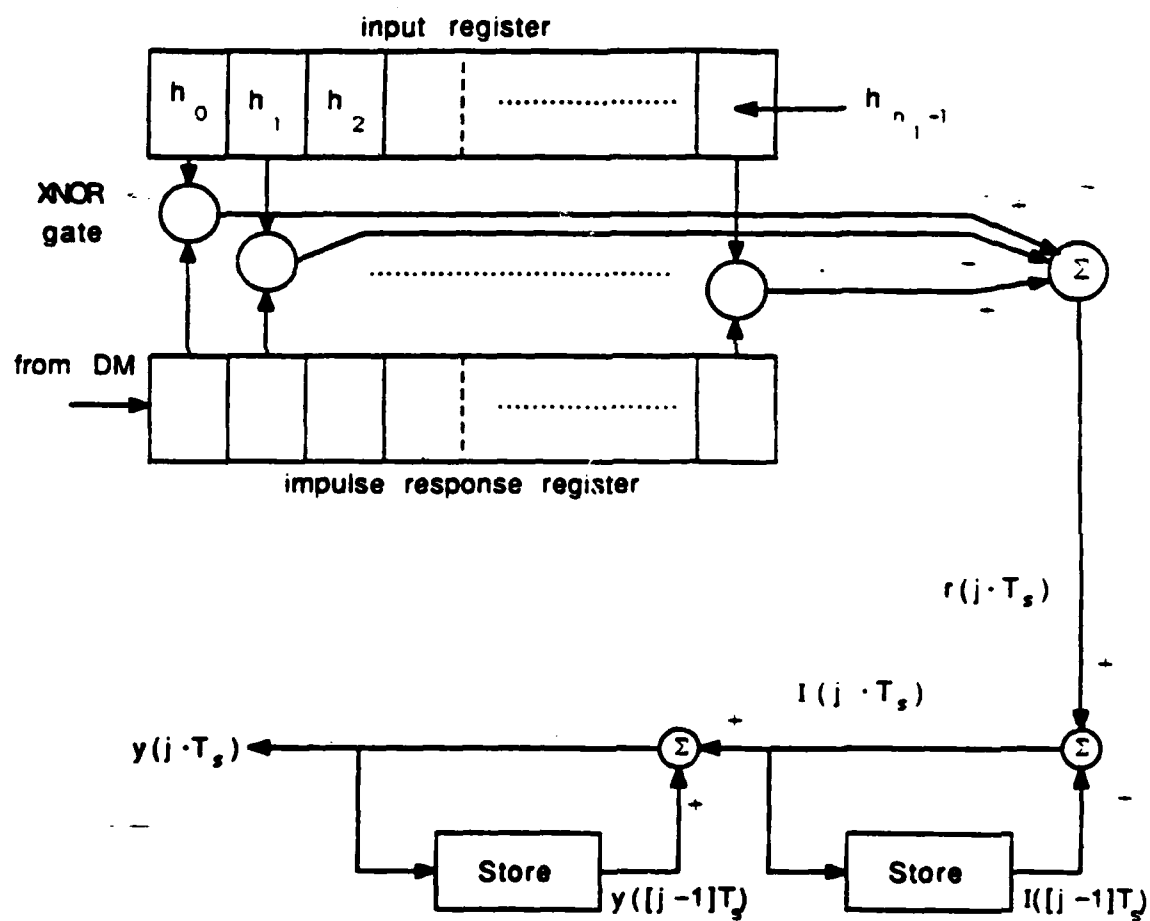
Fig. 4.1. Reduced delta convolution system.

been developed using transient or initial values (while the input register is becoming occupied), it is shown in Appendix B that this same form prevails when the input register is filled with values of $\hat{x}(jT_s)$. So, the output of Fig. 4.1 represents the convolution of $x(t)$ and $h(t)$ when the input register is fully occupied, and the output is a transient response when the register is not fully occupied (loading and unloading). This transient behavior is part of any filter realization that uses delay elements (digital or recursive filters).

A. CIRCUIT REALIZATION OF TRANSVERSAL FILTERS USING THE RDC METHOD

This section first considers a circuit based on Equation (B.11) which functions as a convolver (linear filter) and then a similar circuit which is a correlator. We show that a convolver or a correlator can be constructed by using two shift registers, n_1 XNOR gates, one or two amplifiers, one or two summers and two feedback loops. We show that by simply reversing the contents of the impulse response register, a convolver, in essence, becomes a correlator.

1. RDC Transversal Filter as a Convolver

From the preceding discussions and Appendix B, Fig. 4.2 is a diagram of a system which can be used to implement a RDC transversal filter which is a convolver.

In the realization of the filter of Fig. 4.2, we show the necessary delays as consisting of charge-coupled devices (CCD). Not shown in Fig. 4.2 is the clock of rate $\frac{1}{T_s}$ ticks/second used to transport the bit stream through register R_x and used to transport the multilevel voltages through the CCD.

61
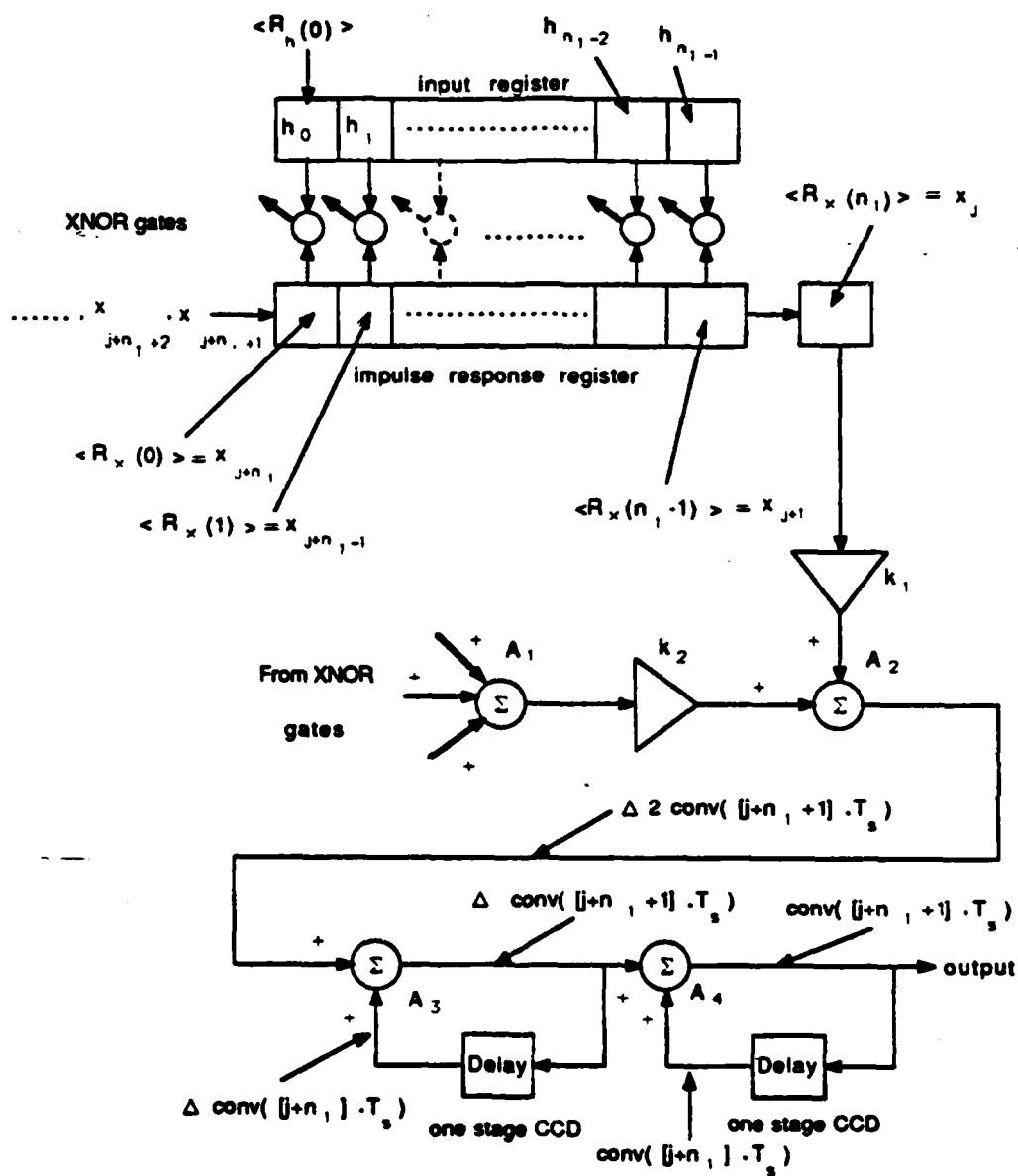
Fig. 4.2. RDC transversal filter convolver.

In Fig. 4.2,

$$conv(j \cdot T_s) = \text{output} = y(j \cdot T_s) \approx y(t)$$

$$\Delta conv(j \cdot T_s) \triangleq conv(j \cdot T_s) - conv([j-1] \cdot T_s)$$

$$\Delta 2conv(j \cdot T_s) \triangleq \Delta conv(j \cdot T_s) - \Delta conv([j-1] \cdot T_s).$$

In Appendix B, it is shown that

$$\Delta 2conv[j \cdot T_s] = C_1 \cdot \left\{ \sum_{i=0}^{n_1-1} < R_h(i) > \cdot < R_x(i) > - < R_x(n_1) > \cdot \sum_{i=0}^{n_1-1} h_i \right\}$$

where $< R_h(i) >$ represents the contents (± 1) of cell i of the impulse response register, etc., and where $< R_x(i) >$ changes with time. So, the value of $\Delta 2conv[j \cdot T_s]$ depends on the contents of register R_x at time $j \cdot T_s$.

There are several constants in Fig. 4.2 which are defined as follows:

$$k_1 = \text{amplifier voltage gain} = -C_1 \cdot \sum_{i=0}^{n_1-1} h_i$$

$$k_2 = \text{amplifier voltage gain} = C_1$$

A_1, A_2, A_3 and A_4 are voltage summers.

2. RDC Transversal Filter as a Correlator.

When the RDC system is used as a correlator, then the content of the impulse response register is reversed as shown in Chapter 2. Following the same procedures given in Appendices A and B, the following result is obtained

$$\Delta 2corr(j \cdot T_s) = -C_1 \cdot \left\{ \sum_{i=0}^{n_1-1} < R_h(i) > \cdot < R_x(i+1) > - < R_x(0) > \cdot \sum_{i=0}^{n_1-1} h_i \right\}$$
(4.2)

where

$$\Delta 2corr(j \cdot T_s) \triangleq \Delta corr(j \cdot T_s) - \Delta corr([j-1] \cdot T_s)$$

$$\Delta corr(j \cdot T_s) \triangleq corr(j \cdot T_s) - corr([j-1] \cdot T_s)$$

63

and where $\Delta 2corr(j \cdot T_s)$, $\Delta corr(j \cdot T_s)$ and $corr(j \cdot T_s)$ are the incremental change in the change of the discrete correlation values, the change in the discrete correlation values, and the result of correlation at time $j \cdot T_s$.

Fig. 4.3 is a block diagram of a correlator as given in Equation (4.2) where

$$k_3 = C_1 \cdot \sum_{i=0}^{n_1-1} h_i = -k_1 \text{ and } k_4 = -C_1 = -k_2.$$

As expected, there is similarity of structure and commonality of hardware in the realizations of the convolver and correlator. In fact, if the input register is bidirectional and if the amplifiers become inverting, then a convolver becomes a correlator and vice versa.

We can draw some conclusions concerning hardware from Fig. 4.2.

a. When $h(t)$ begins and ends at 0 volts, then*

$$\sum_{i=0}^{n_1-1} h_i = 0$$

and we will not need the amplifier of gain k_1 nor the summer A_2.

b. The needed storage equals $(2n_1 + 1)$ binary stages or cells and 2 single stage CCD's.

c. The needed number of binary multipliers (XNOR gates) is n_1.

d. The needed number of operational amplifiers is 4.

e. C_1 is a scale factor (could be apart from the IC realization of the RDC system).

f. The same IC can be used as a convolver or a correlator.

---

* Given $h(n_1 \cdot T_s) = 0$. We know $h_s(n_1 \cdot T_s) \approx h(n_1 \cdot T_s)$. But $h_s(n_1 \cdot T_s) = \Delta h \cdot \sum_{i=0}^{n_1-1} h_i$ when $h(0) = 0$. So, when $h(n_1 \cdot T_s) = 0$, then $\sum_{i=0}^{n_1-1} h_i = 0$, under the condition $h(0) = 0, (h_s(0)$ is always 0$)$.
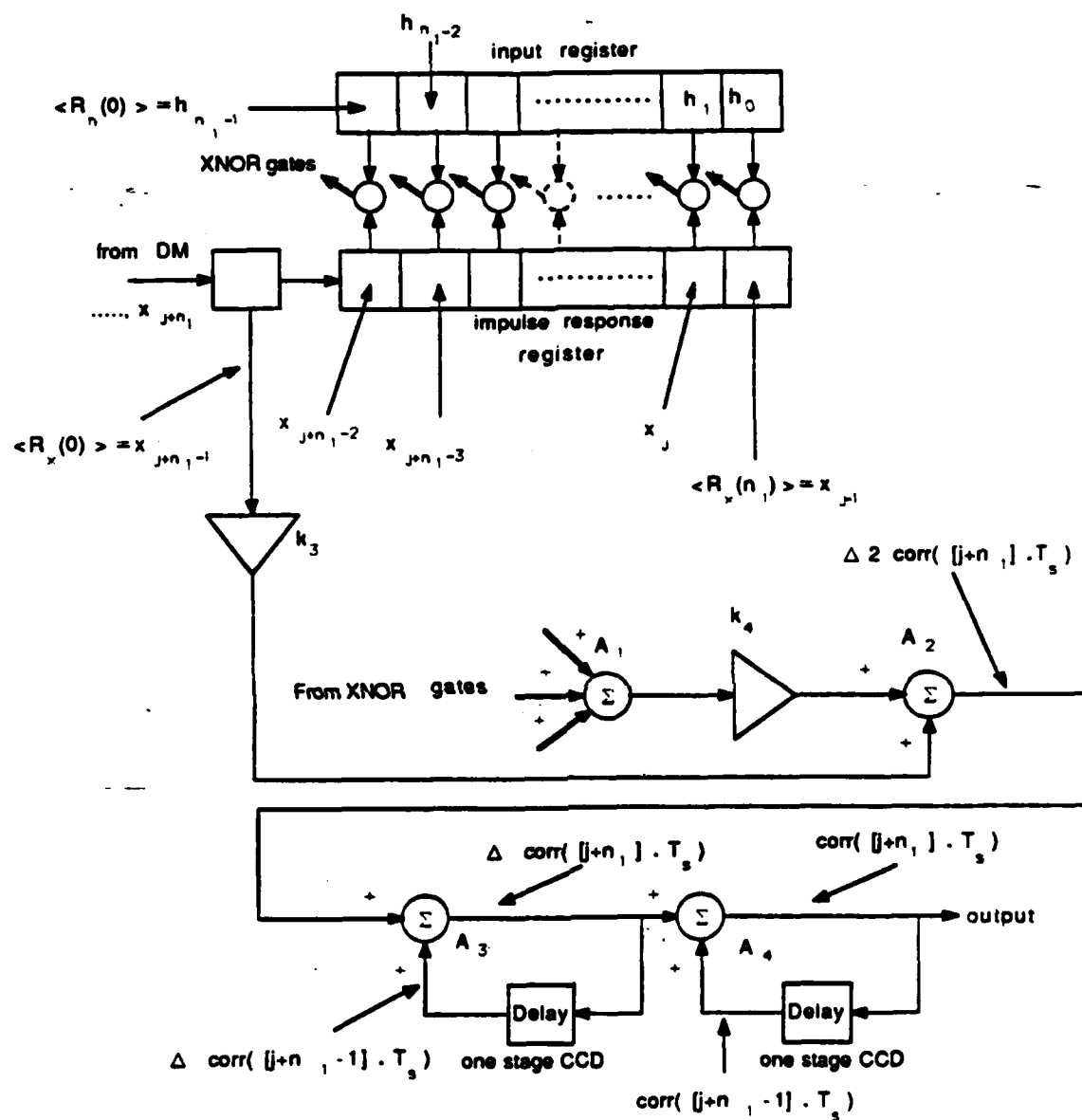
Fig. 4.3. RDC transversal filter correlator.

We can now identify some important features of the RDC system.

    a. Uses digital technology (compatible with VLSI).

    b. There are no code words and hence no synchronization requirements.

    c. There are no multipliers (XNOR gates accomplish an equivalent function).

    d. Real time operation (no off-line processing).

    e. Easily programmable to represent impulse response of interest.

    f. No limitation on the time duration T_d of the signal in matched filter applications.

    g. No theoretical limitation on the number of register stages (taps); filter sections can be easily cascaded using several IC's to realize longer duration impulse responses.

    h. Reduced hardware requirements (complexity) compared with other FIR filter realizations. An example is provided in Chapter 5.

In Chapter 7, we conclude from simulation results that when used as a matched filter or ordinary linear filter, the RDC system has noise performance which is better than predicted by ordinary filter theory. The reasons for this are explored in Chapter 7.

This chapter (4) presents a system (block diagram) which realizes an approximation to the convolution integral. The system can be used as a linear filter or a correlator. The next chapter details the hardware (circuit diagram) required to realize the RDC system.

66

V. HARDWARE REALIZATION OF THE RDC SYSTEM

A. INTRODUCTION

In this chapter we introduce a possible circuit to implement the RDC system. We also compare the needed hardware and complexity of the RDC system and a conventional digital matched filter (DMF). We conclude the RDC system has a simple layout for IC realization and requires less than 10% of the transistors needed for a typical DMF having comparable performance.

B. CIRCUIT DIAGRAM OF THE RDC SYSTEM

Fig. 5.1 shows a possible circuit having as output $\Delta 2conv(j \cdot T_s)$ which is defined in Equation (B.11) and shown in Fig. 4.2. Register R_h contains the bit sequence representing the delta modulation equivalent of the impulse response of the filter being realized. Register R_x contains $(n_1 + 1)$ bits of the delta modulation representation of the input voltage to be filtered. The XNOR gates produce the binary products of the contents of register R_h and register R_x. From Equation (B.11), we need to sum the outputs of the XNOR gates. But, it is difficult to sum voltages. Therefore, we use field-effect transistors (FET's) as voltage-to-current converters and then sum currents. Each FET will be in one of two states:

1) Cut off when its gate has a "−1" (or 0) voltage which means zero current passing through it.

2) Turned on when its gate has a "1" voltage which means I amperes passing through it.
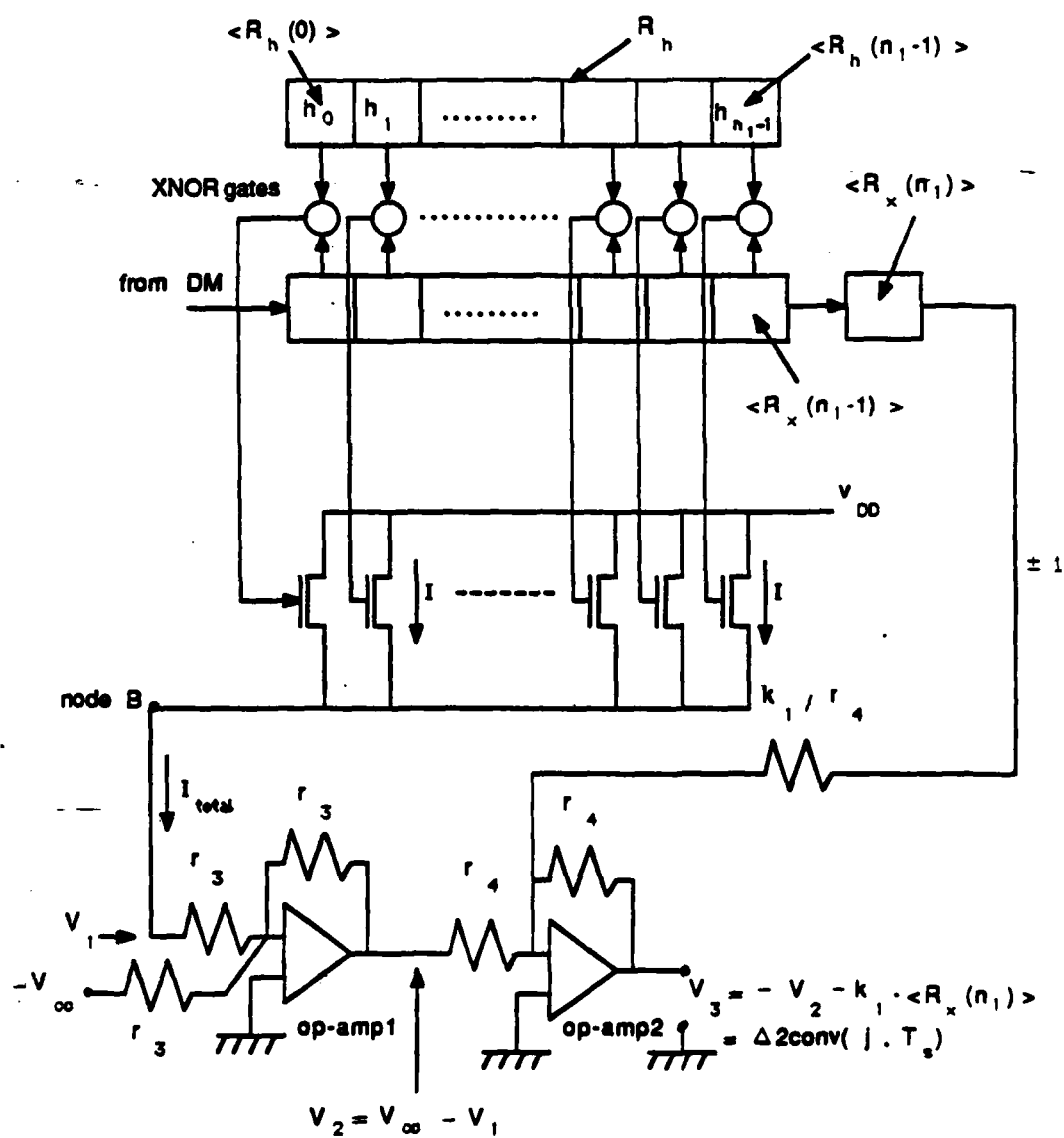
Fig. 5.1. A circuit which provides $\Delta 2conv(j \cdot T_s)$.

68

All the currents from the FET's enter node B to form I_{total} leaving node B. So, from Fig. 5.1,

$$V_1 = r_3 \cdot I_{total},$$

or,

$$V_1 = r_3 \cdot I \cdot (n_e)$$

where $n_e \triangleq$ the number of bits for which $<R_h(i)> = <R_s(i)>; i = 0, 1, 2, \ldots n_1 - 1$

Let $\lambda \triangleq n_e -$ number of bits for which $<R_h(i)> \neq <R_s(i)>; i = 0, 1, 2, \ldots n_1 - 1$

So, $\lambda = n_e - (n_1 - n_e) = 2n_e - n_1; \quad n_e = 0, 1, 2, \ldots n_1.$

As a result of this,

$$V_1 = \frac{r_3 \cdot I}{2}(\lambda + n_1); \qquad \lambda = -n_1, -n_1 + 2, \ldots, n_1 - 2, n_1 \qquad (5.1)$$

and,

$$(V_1)_{max} = r_3 \cdot I \cdot n_1.$$

Assume that $(V_1)_{max} = 2V_{cc}$. Then $V_{cc} = \frac{r_3 \cdot I}{2}$ where V_{cc} is an externally supplied constant (dc) voltage. Using this in Equation (5.1) gives

$$V_1 = \frac{r_3 \cdot I}{2}\lambda + V_{cc} \qquad (5.2)$$

Fig. 5.2 shows V_1 as a function of λ.

We can use Equation (5.2) to define V_2 in Fig. 5.1 as follows

$$V_2 = V_{cc} - V_1 = V_{cc} - \left[\frac{r_3 \cdot I}{2}\lambda + V_{cc}\right] = -\frac{r_3 \cdot I}{2}\lambda$$

Fig. 5.3 shows $-V_2$ as a function of λ.

Let $k_1 = C_1 \cdot \left(\sum_{i=0}^{n_1-1} h_i\right) =$ constant for the impulse response of interest. From Fig. 5.1

$$V_3 = -V_2 - k_1 \cdot <R_s(n_1)>$$
$$= \frac{r_3 \cdot I}{2}\lambda - k_1 \cdot <R_s(n_1)> \qquad (5.3)$$
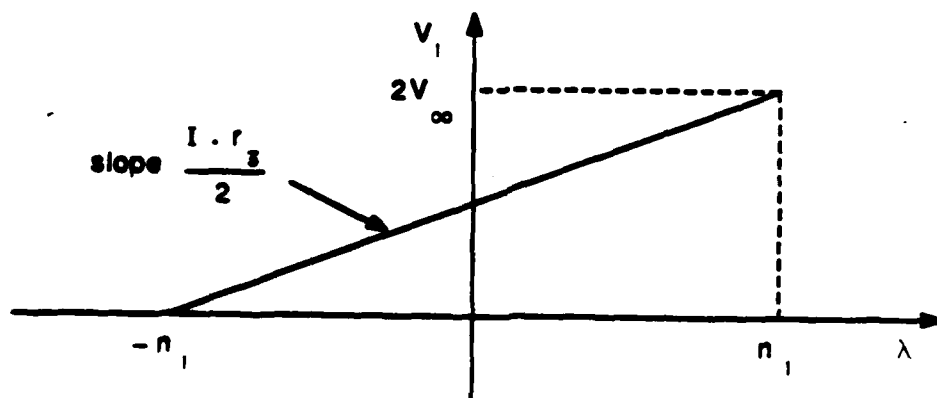
Fig. 5.2. V_1 as a function of λ.
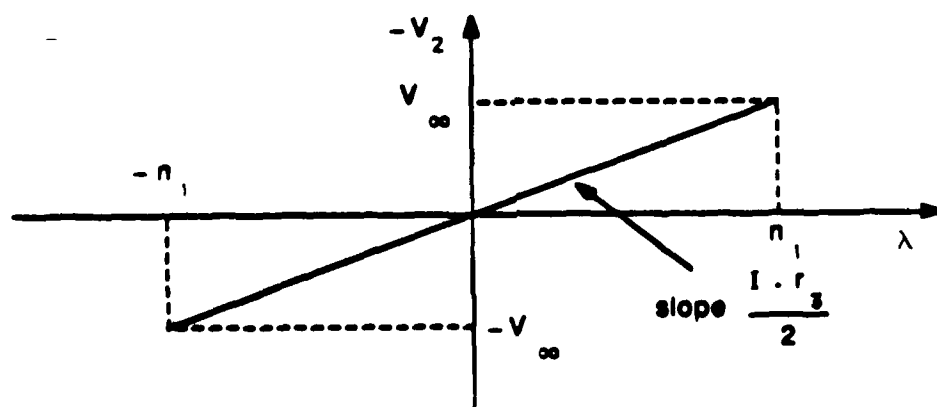
Fig. 5.3. $-V_2$ as a function of λ.

70

From Equation (B.11)

$$\Delta 2conv(j \cdot T_s) = C_1 \cdot \left\{ \sum_{i=0}^{n_1-1} <R_h(i)> \cdot <R_x(i)> \right\} - k_1 \cdot <R_x(n_1)>$$

But,

$$\sum_{i=0}^{n_1-1} <R_h(i)> \cdot <R_x(i)> = \lambda$$

Therefore, we can write

$$\Delta 2conv(j \cdot T_s) = C_1 \cdot \lambda - k_1 \cdot <R_x(n_1)> \cdot \qquad (5.4)$$

From Equations (5.3) and (5.4), we can make $V_3 = \Delta 2conv(j \cdot T_s)$ if we set $C_1 = \frac{r_3 \cdot I}{2}$.

Knowing the step sizes Δx and Δh, the sampling frequency $\frac{1}{T_s}$, and the impulse response register contents and length n_1, we can determine the constants C_1 and k_1. Knowing n_1, C_1 and choosing a convenient value for I, we can determine r_3 and V_{cc}. We choose V_{DD} in Fig. 5.1 sufficiently larger than $(V_1)_{max}$ so that the current I is constant.

The circuit of Fig. 5.1 has as its output $\Delta 2conv(j \cdot T_s)$. To find $y(j \cdot T_s)$, the value of convolution, we have to use two feedback circuits as mentioned in Chapter 4. Fig. 5.4 shows one circuit which provides $y(j \cdot T_s)$ when its input is $\Delta 2conv(j \cdot T_s)$. The values of r_5 are arbitrary. From Fig. 5.4,

$$y(j \cdot T_s) = conv(j \cdot T_s)$$
$$= conv((j-1) \cdot T_s) + \Delta conv(j \cdot T_s)$$

71
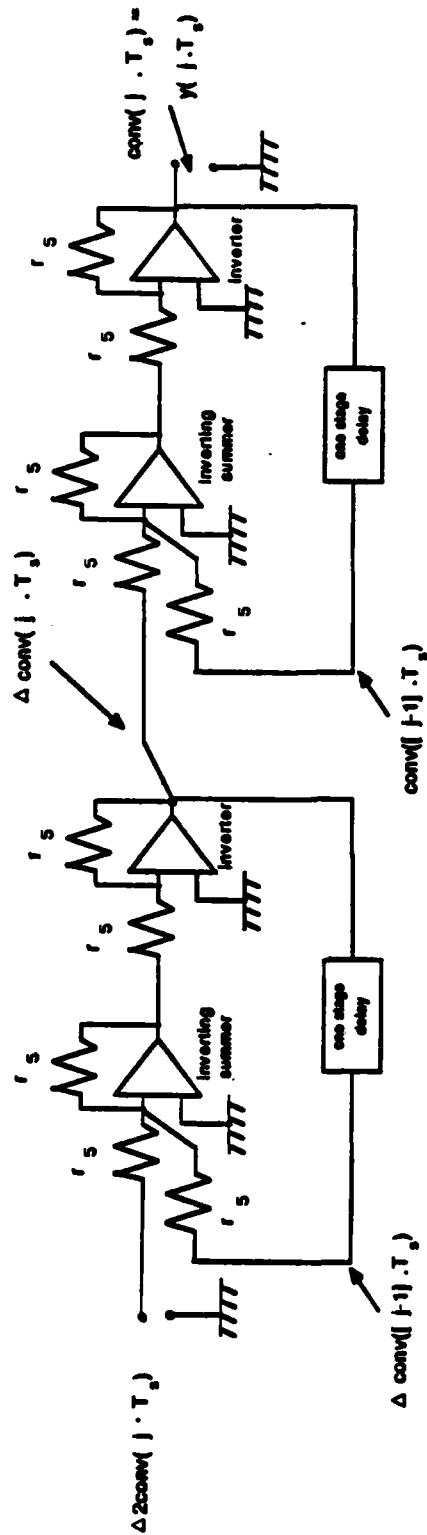
Fig. 5.4. A diagram to find the delta convolution.

where

$$\Delta conv(j \cdot T_s) = \Delta conv((j-1)T_s) + \Delta 2conv(j \cdot T_s)$$

and $\Delta 2conv(j \cdot T_s)$ is given in Equation (5.4).

C. HARDWARE COMPARISON OF THE RDC SYSTEM AND A TYPICAL DIGITAL MATCHED FILTER (DMF)

We can estimate the hardware requirements of the RDC system using Figures 5.1 and 5.4. There are four different circuits identified in those figures. There is also a delta modulator which could be part of the system or could be off the chip when using IC technology.

Assume 6 transistors are needed to realize each stage of a shift register. Then the RDC circuit requires $6n_1$ transistors for the impulse response register and $6(n_1 + 1)$ for the input register for a total of $6(2n_1 + 1)$ transistors.

Assume each XNOR gate requires 10 transistors. Then the array of gates requires $10n_1$ transistors.

There are n_1 field-effect transistors.

Assume 20 transistors are required for each op-amp for a total of 120 transistors.

The total hardware requirements, neglecting resistors and capacitors, is $(23n_1 + 126)$ transistors plus 2 single stage analog storage elements (CCD's) and possibly a delta modulator. The hardware (in fact, the circuitry) is the same for the realization of a convolver (lowpass filter, bandpass filter, etc.) as for a correlator (matched filter).

73

A conventional way to implement a DMF is presented by Turin [10] (see Appendix C). We will compare the hardware required for this DMF with that previously tabulated for the RDC system.

Assume that the sampling frequencies for the conventional method given by Turin [10] and the RDC method are $2f_{max}$ and $2 \cdot F \cdot f_{max}$ respectively, where f_{max} is the maximum frequency component of the band limited version of the input signal and F is the oversampling ratio. As a result of this

$$\frac{n_1}{L} = \frac{2 \cdot F}{2} = F$$

where n_1 is the register length of the RDC system and L is that of the conventional DMF register.

We can now list the needed hardware to implement a conventional DMF and that needed to implement an equivalent RDC filter. The results are in Table 5.1.

If we assume each adder needs only 6 transistors, then the needed transistors to implement the DMF filter, neglecting the D/A and A/D converters and the amplifiers required are

$$6(N + M) \cdot L + 10M \cdot N \cdot L + 6M \cdot N(L - 1)$$

where the needed transistors for the RDC system are

$$23n_1 + 126 = 23(F \cdot L) + 126$$

Define R_T as

$$R_T = \frac{\text{the needed transistors when we use RDC method}}{\text{the needed transistors when we use DMF method}}$$

74

TABLE 5.1. COMPARISON BETWEEN THE DMF CONVENTIONAL METHOD AND THE RDC METHOD

Conventional DMF	RDC
1. N shift registers having $N \cdot L$ binary stages and one having $M \cdot L$ binary stages	Two shift registers having $2n_1 + 1$ binary stages
2. $M \cdot N \cdot L$ binary multipliers (XNOR gates)	n_1 binary multipliers (XNOR gates)
3. $M \cdot N \cdot (L-1)$ summers	6 operational amplifiers, each having less than 20 transistors.
4. A/D converter	Delta modulator
5. D/A converter	none required
6. $M \cdot N$ sets of interconnections where the number of these sets equals to L	n_1 interconnections

For large values of L, then

$$R_T \approx \frac{23F}{6(N+M) + 16M \cdot N} \tag{5.5}$$

A common value for M and N is 8

$$\text{then } R_T \approx \frac{23F}{1120} \tag{5.6}$$

From Equation (5.6) we conclude that the circuit complexity is equal for $F \approx 50$.

When $L = 1000$, $M = N = 8$, and $F = 4$ then $R_T = 0.0823$. In general, we conclude that the number of transistors needed to implement the RDC method

is less than 10% of that needed to implement the conventional DMF filter. In addition the circuit layout of a conventional DMF method is more complex than that of the RDC system.

VI. PERFORMANCE OF THE RDC METHOD

In this chapter we present simulation results which show the transfer function of a RDC lowpass filter (LPF), the response of a RDC LPF to two signals, one in the pass band and the other outside the pass band, and the output of a RDC LPF with a square wave input. Included are responses of filters matched to a raised cosine pulse and to a chirp signal.

A. LOWPASS FILTER PERFORMANCE

1. Transfer Function of the RDC System

Fig. 6.1 is a block diagram of the RDC system used to find the transfer function when used as a LPF. We assume a LPF cutoff frequency f_c of 1.0 Hz. The impulse response $h(t)$ is, then,

$$h(t) = 2 \cdot f_c \cdot \frac{\sin(2\pi f_c t)}{2\pi f_c t} = 2\frac{\sin(2\pi t)}{2\pi t}$$

where the corresponding causal impulse response $h_c(t)$ together with its staircase representation $h_{cs}(t)$ are plotted in Fig. 6.2.

We obtain the transfer function by applying to the RDC system the signal

$$x(t) = A_i \cdot \sin(2\pi f_i t):$$

Following the transient time, which is dependent upon the register length n_1 and the sampling interval T_s, the output $y(t)$ is

$$y(t) = A_0 \cdot \sin(2\pi f_i t + \phi) \text{ where } \phi \approx 0.$$

The ratio $\frac{A_0}{A_i}$ as a function of frequency is taken as the magnitude of the RDC LPF transfer function.

77
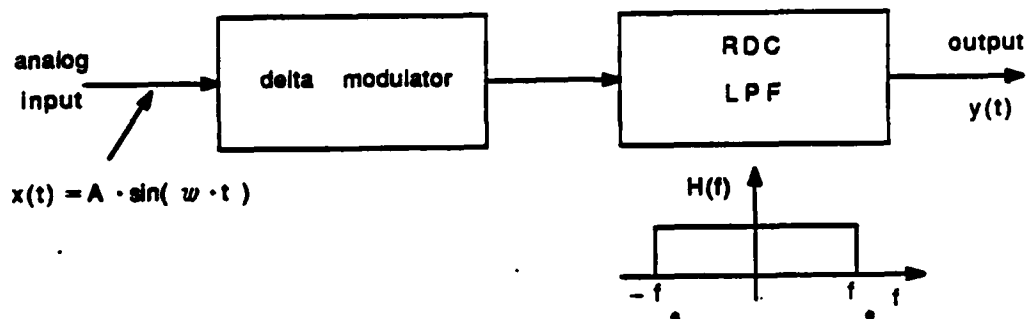
Fig. 6.1. Block diagram of the system
used to find the RDC LPF transfer function.
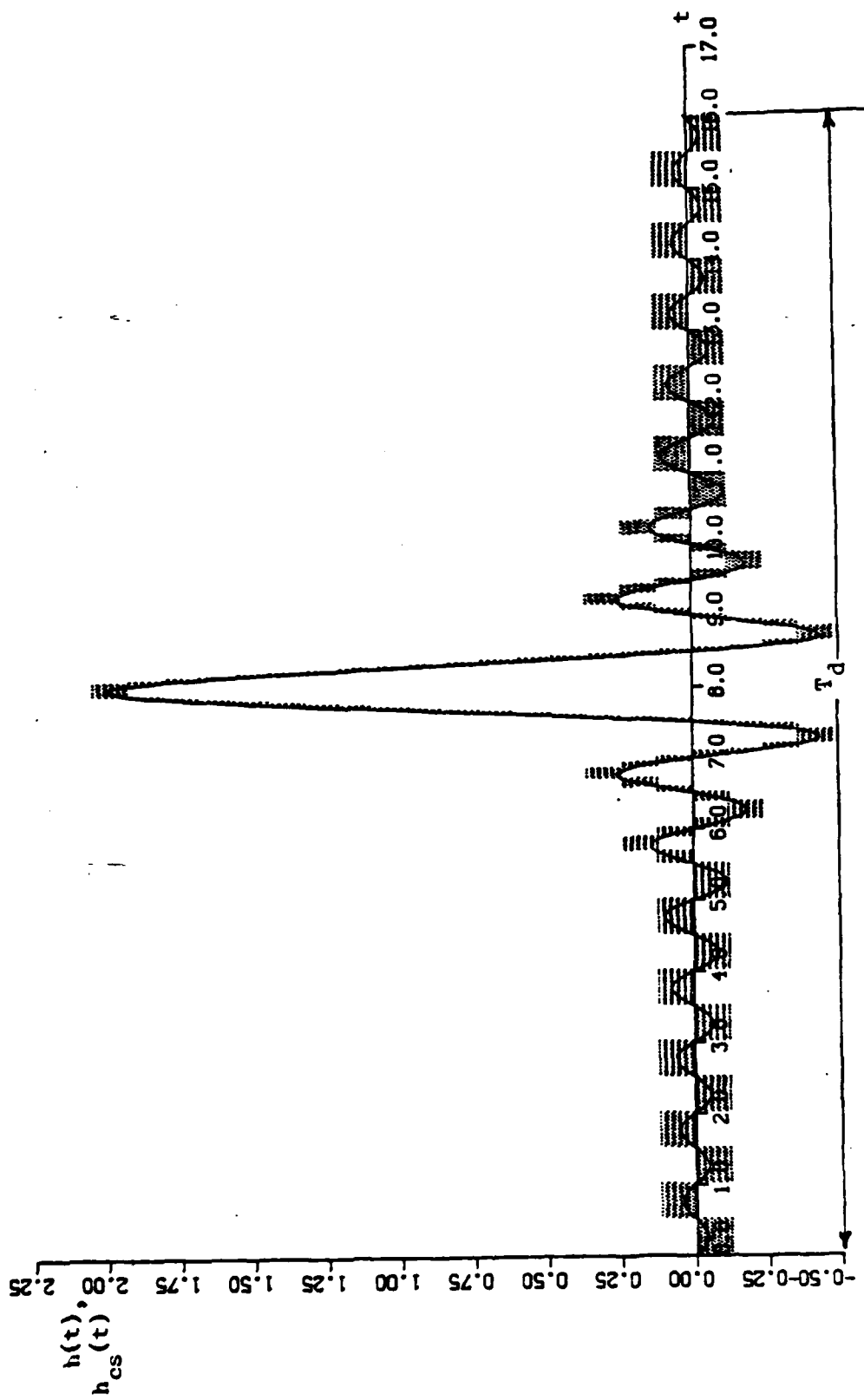
Fig. 6.2. The causal transfer function for the LPF $h_c(t)$ together with its staircase representation $h_{cs}(t)$.

The register length n_1 depends on T_s and T_d which is shown in Fig. 6.2. In fact, $n_1 = \frac{T_d}{T_s}$. For a fixed T_s, increasing n_1 permits a more inclusive representation of the impulse response $h(t)$. In the following, we measure T_d in terms of the number of zeroes about the main lobe of $h_c(t)$ and included in T_d.

Figures 6.3, 6.4, and 6.5 show the magnitude of the transfer function for the RDC LPF where the number of zeroes of the impulse response are 2, 6, and 8 respectively. As expected, the LPF transfer function "improves" as more zeroes of $h_c(t)$ are included.

Fig. 6.6 shows the magnitude of the transfer function of the RDC LPF when T_d includes 6 zeroes of $h_c(t)$ and when rectangular and Hamming windows are used. We see that the ripple in the transfer function is reduced using the Hamming window.

When the step size Δh becomes larger than the side lobes of the impulse response, then the delta modulator considers the side lobes are of equal amplitudes (see Fig. 6.2). To alleviate this problem we can choose a smaller step size and increase the sampling rate to maintain the DM slope-following capacity. This results in increased register length n_1. An alternative is use of a variable step size RDC system which is treated in Chapter 7.
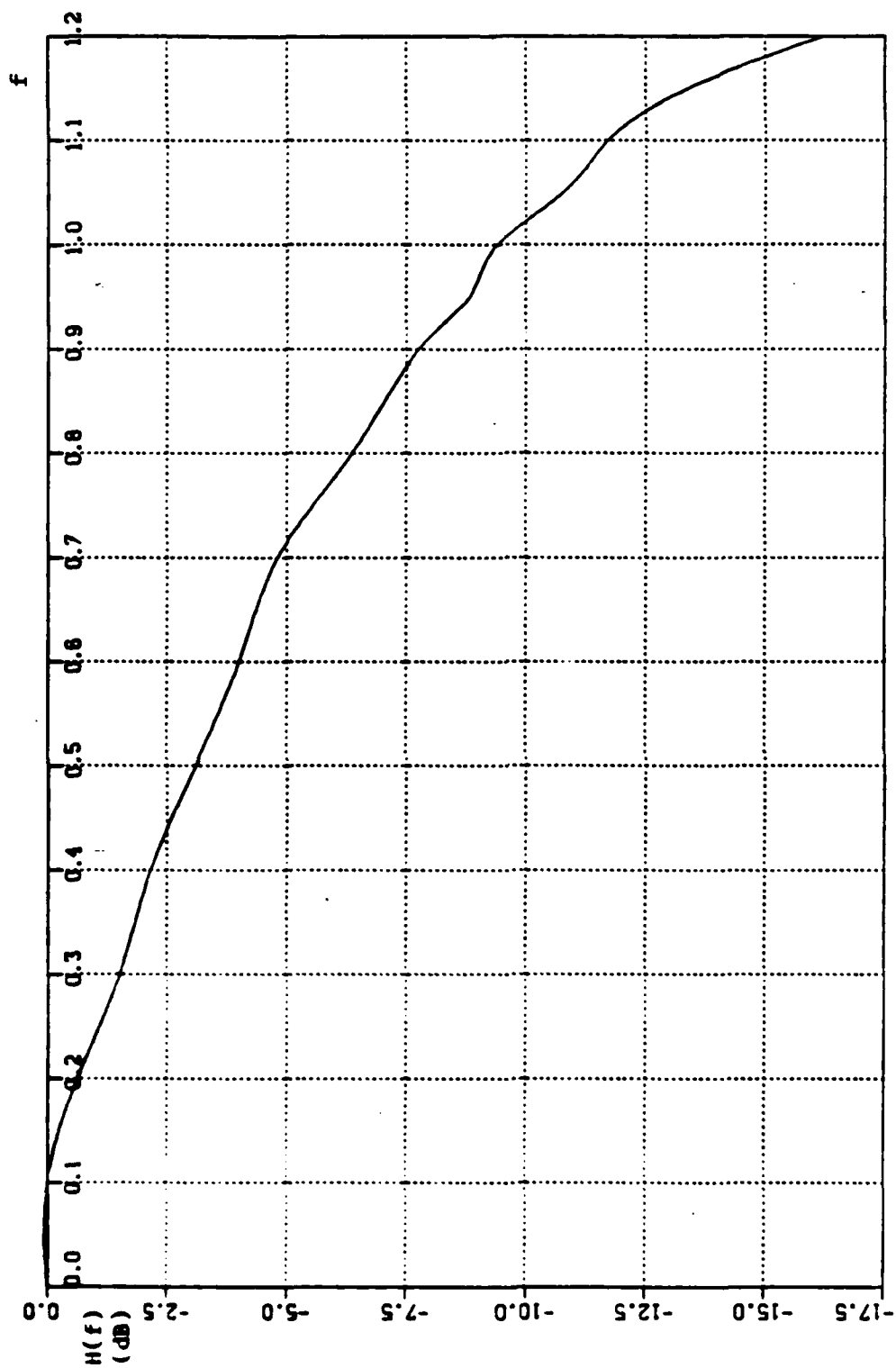
Fig. 6.3. The RDC LPF transfer function when T_d includes 2 zeroes of $h_c(t)$.
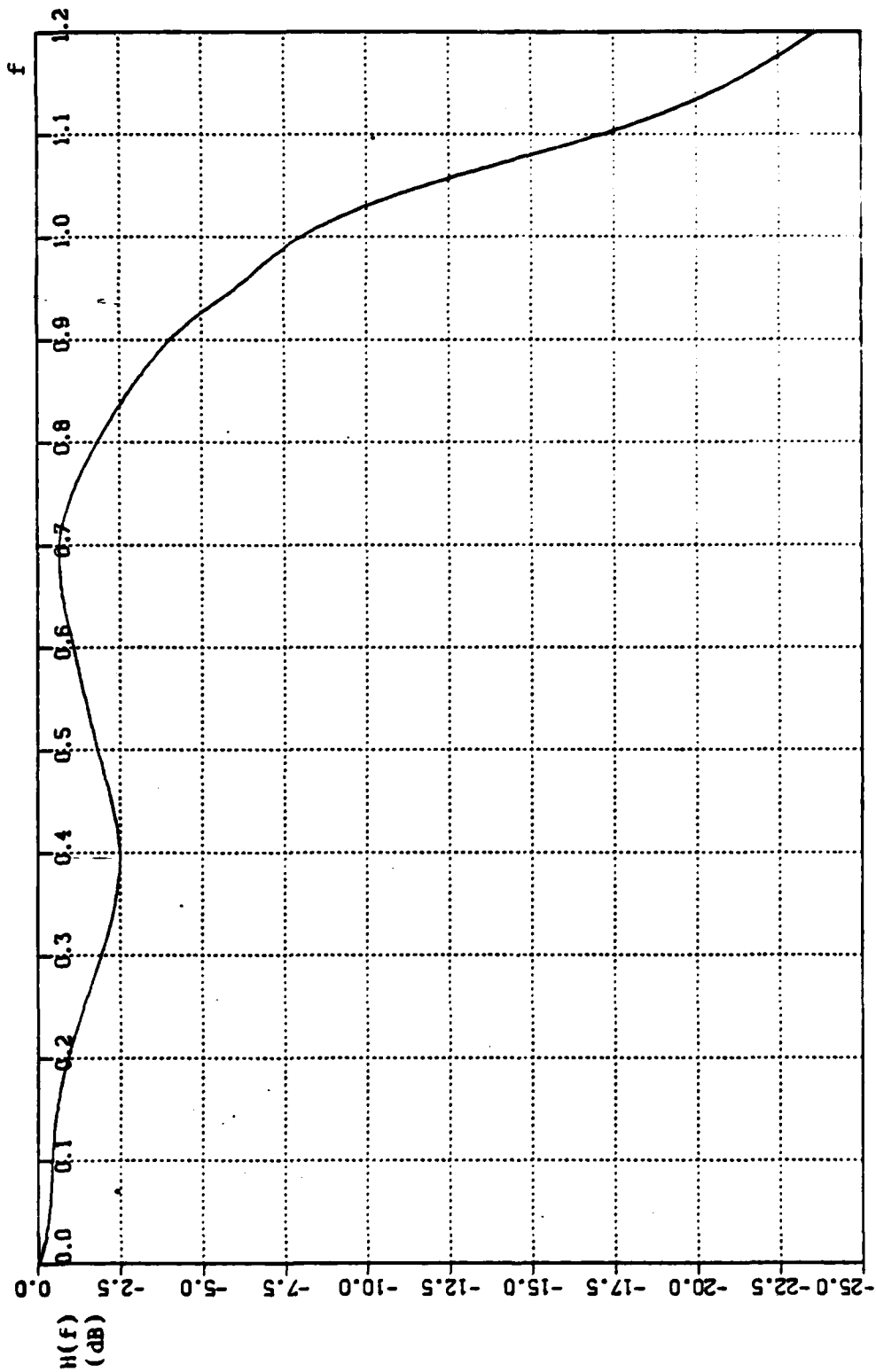
Fig. 6.4. The RDC LPF transfer function when T_d includes 6 zeroes of $h_c(t)$.

82
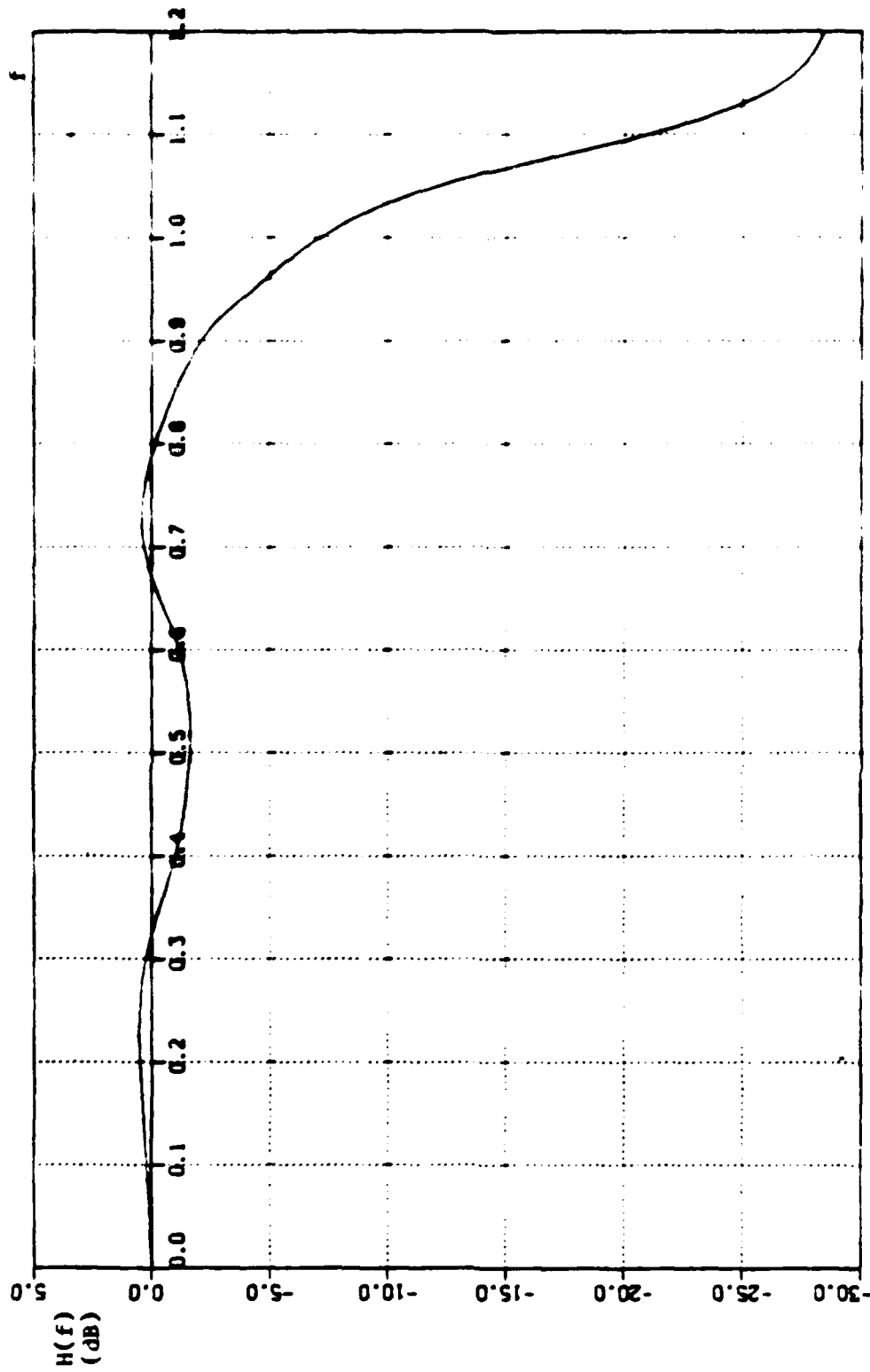
Fig. 6.5. The RDC LPF transfer function when T_d includes 8 zeros of $h_c(t)$.
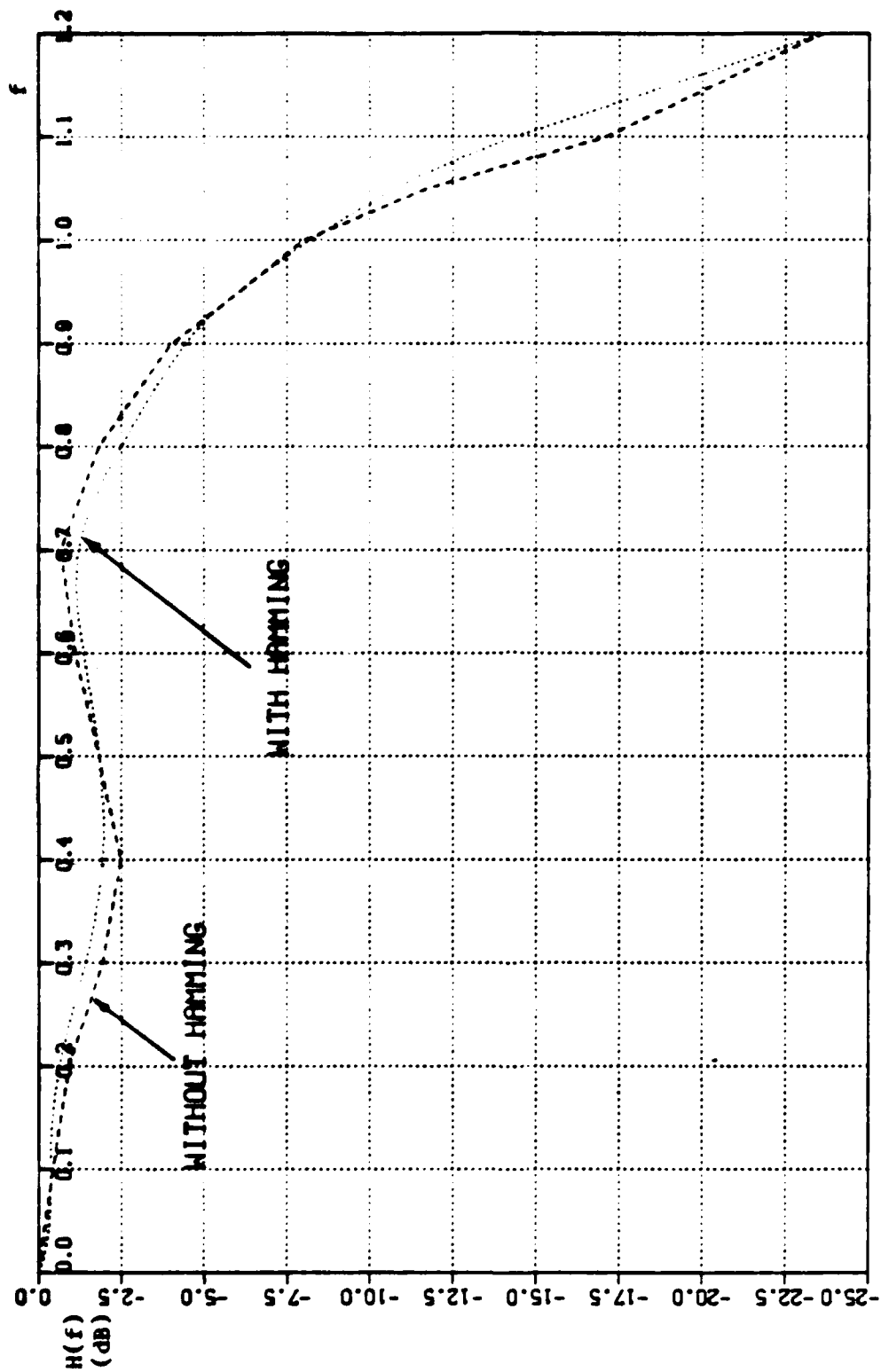
Fig. 6.6. The RDC LPF transfer function when T_d includes 6 zeroes of $h_c(t)$ and when rectangular and Hamming windows are used.

84

2. The Output of the RDC LPF When the Input is the Sum of Two Sinusoids, One in the Pass Band and the Other Outside the Pass Band

Assume that the RDC LPF with $f_c = 1.0$ Hz, has an input

$$x(t) = 0.5 \sin\left(\frac{w_1 t}{3}\right) + 1.0 \sin(w_1 t)$$

where $w_1 = 2\pi f_1$ and $f_1 = 1.2$Hz.

Figures 6.7 and 6.8 show the input $x(t)$, its staircase representation $x_s(t)$, and the output $y(t)$ of the filter. We see that the 0.4 Hz sinusoid is preserved in the output while the 1.2 Hz term is suppressed. The RDC system when configured as a LPF does discriminate (filter) on the basis of frequency.

3. The Output of a RDC LPF with a Square Wave Input

Assume $f_c = 1.0$ Hz. Assume the input $x(t)$ is a square wave of frequency $f_0 = 0.25$ Hz, where Fig. 6.9 shows the input $x(t)$ and its staircase representation $x_s(t)$. It is expected that the first and third harmonics of the input will appear in the output of the LPF and all other harmonics will be suppressed. Further, the amplitude of the third harmonic should be one third that of the first harmonic according to the Fourier series representation of the square wave. The simulation result agrees with this expectation as shown in Fig. 6.10.

Assume that the square wave input has a fundamental frequency of 0.5 Hz. Fig. 6.11 shows that all harmonics are suppressed in the output of the RDC LPF, as desired.
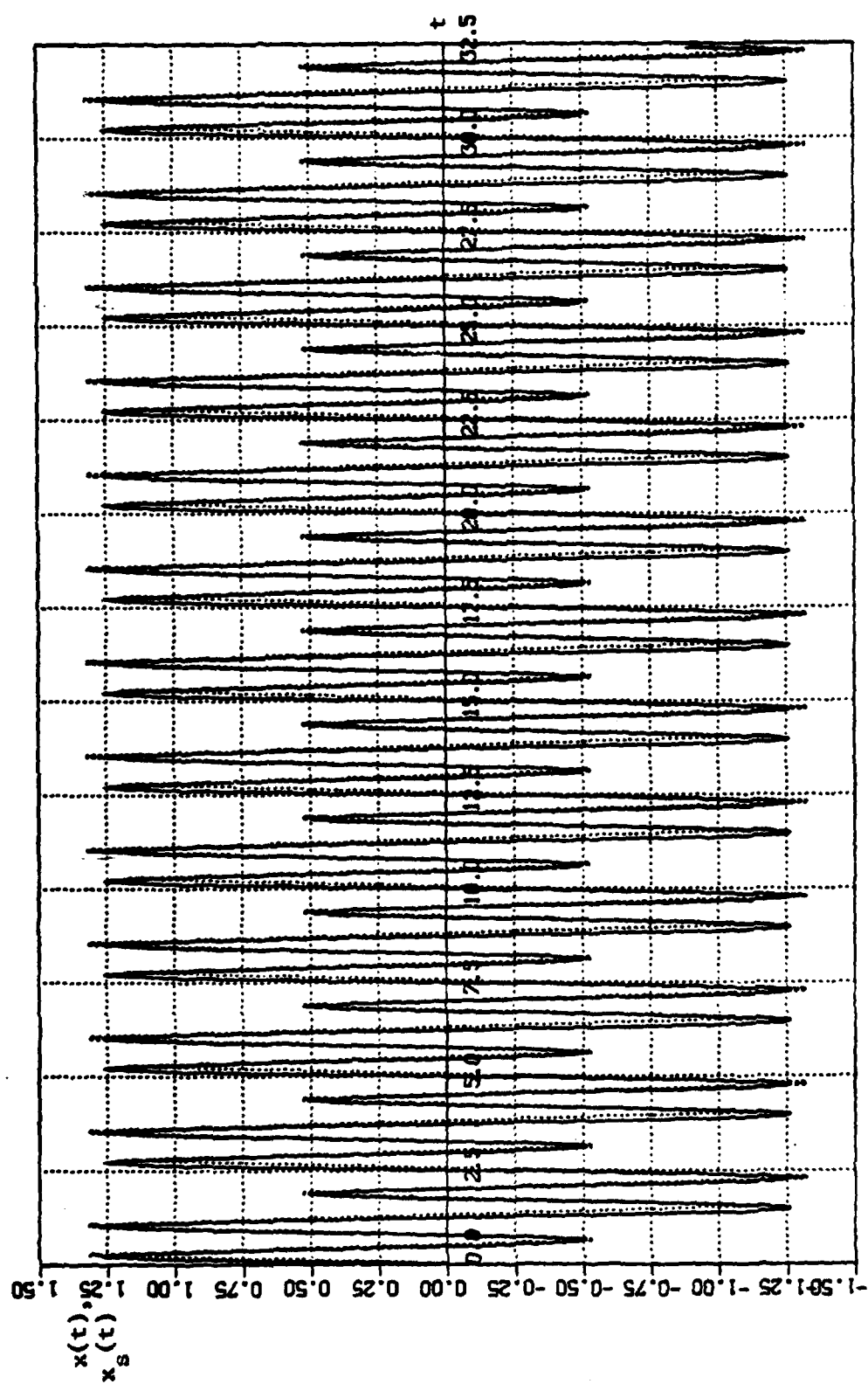
Fig. 6.7. The input $x(t)$ and its staircase representation $x_s(t)$.

86
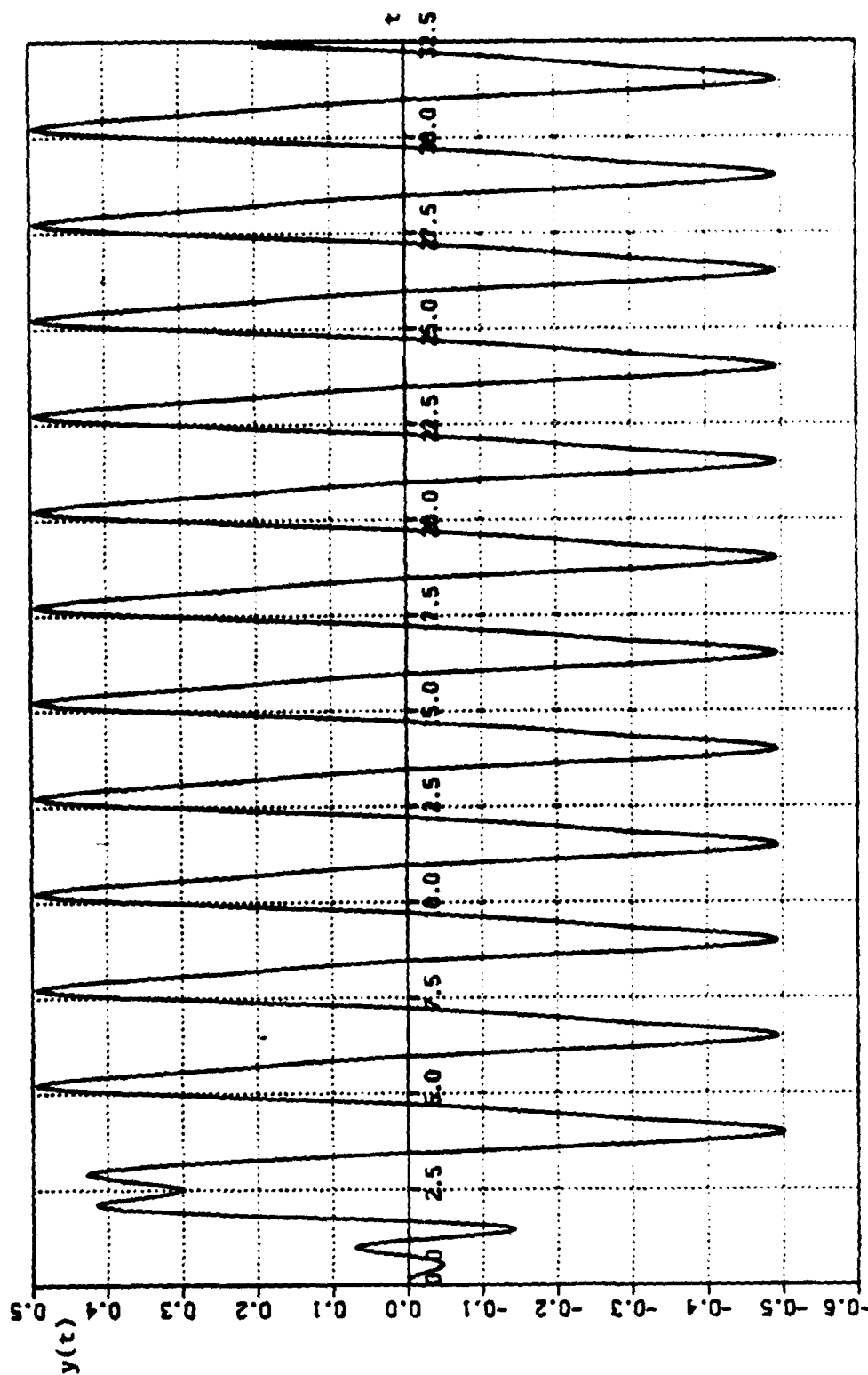
Fig. 6.8. The output of the RDC LPF when the input is the sum of two sinusoids, one in the pass band and the other outside the pass band.

87
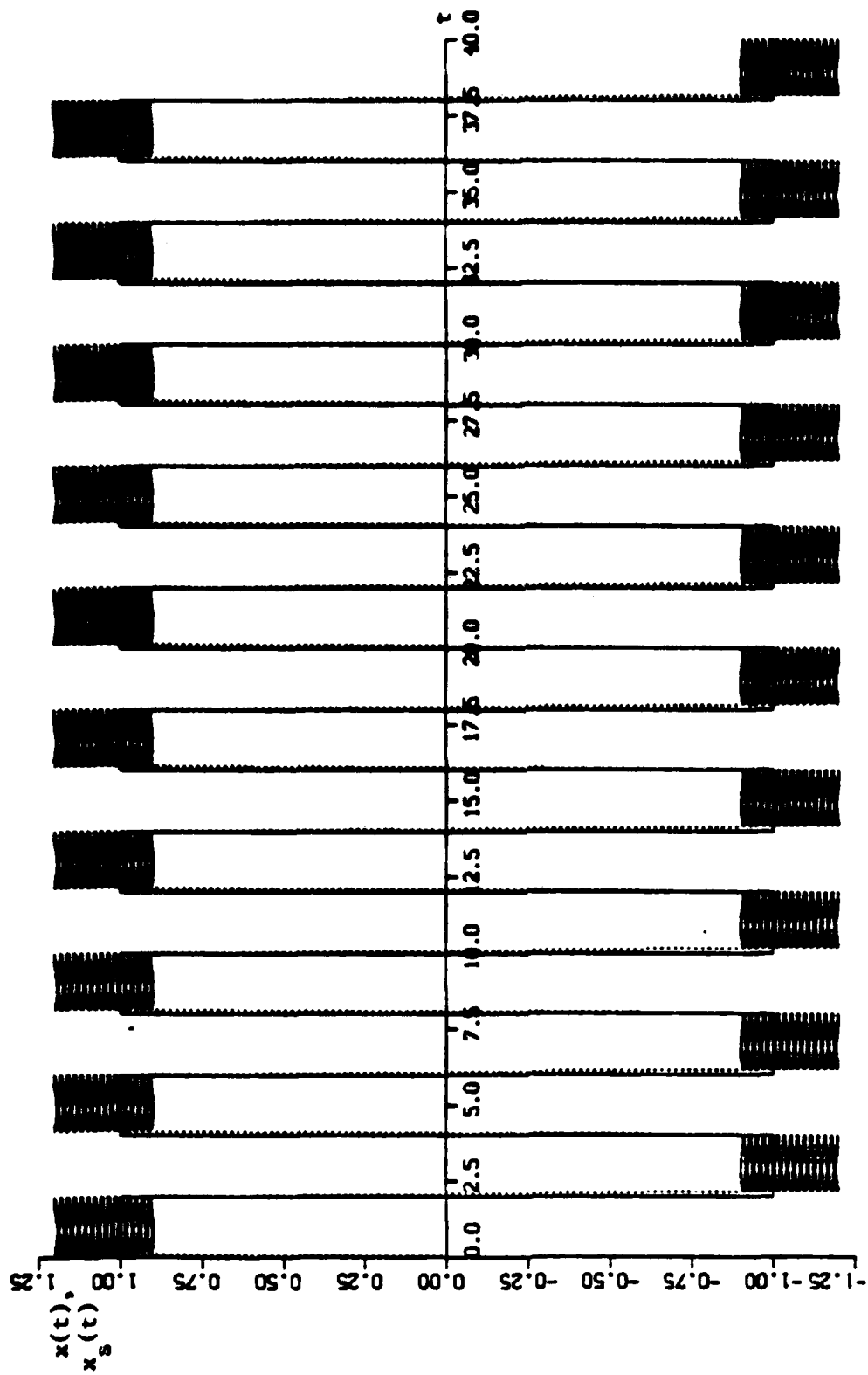
Fig. 6.9. Input square wave $x(t)$ and its staircase representation $x_s(t)$.
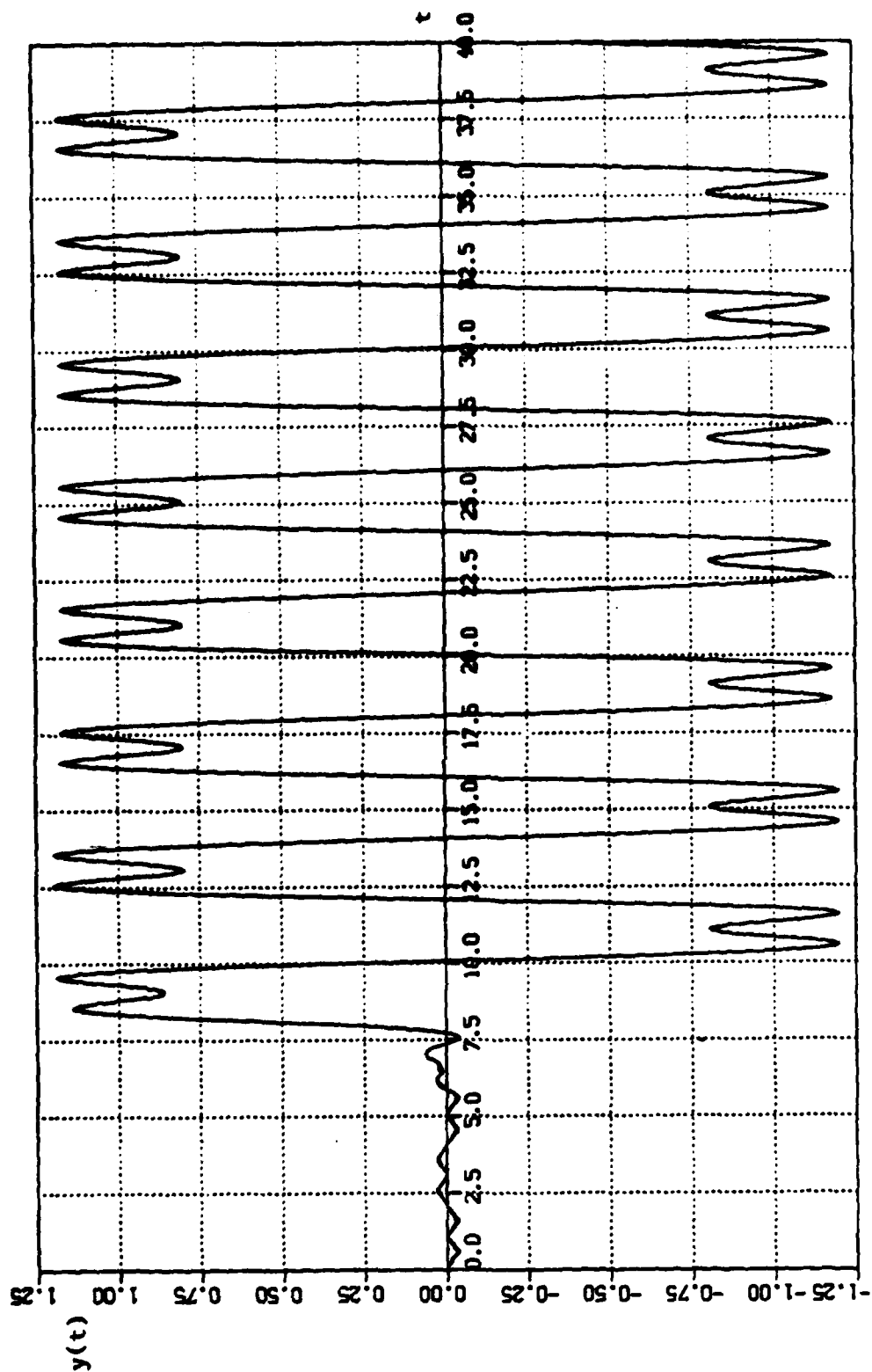
Fig. 6.10. The output of the RDC LPF when the input is a square wave having a fundamental frequency of 0.25 Hz.

89
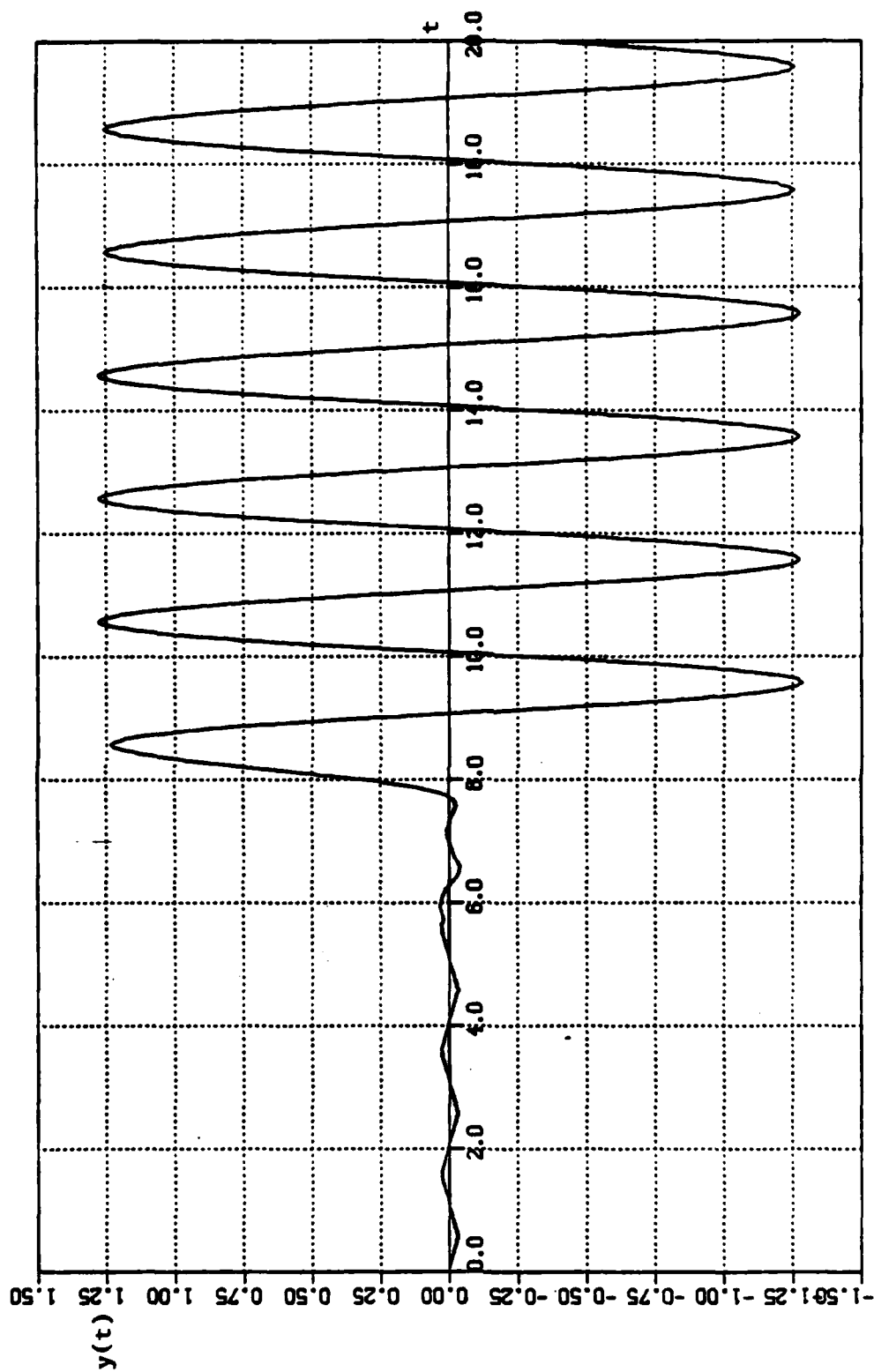
Fig. 6.11. The output of the RDC LPF when the input is a square wave having a fundamental frequency of 0.5 Hz.

B. MATCHED FILTER PERFORMANCE

1. Filter Matched to a Raised Cosine Pulse

Fig. 6.12 shows a block diagram of the system used to simulate the implementation of a filter matched to a raised cosine signal. The impulse response of the matched filter

$$h(t) = x(T_d - t) = \begin{cases} A\left(1 + \cos\left(2\pi\left(\frac{2t}{T_d} - \frac{1}{2}\right)\right)\right) & 0 \le t \le \frac{T_d}{2} \\ -A\left(1 + \cos\left(2\pi\left(\frac{2t}{T_d} + \frac{1}{2}\right)\right)\right) & \frac{T_d}{2} \le t \le T_d \end{cases}$$

where $x(t)$ is the input signal and T_d is the time duration. Fig. 6.13 shows the raised cosine impulse response where A was chosen to be equal to 1.0 volt and $T_d = 1.0$ second.

Figures 6.14 and 6.15 show the results of correlation when we use the conventional methods $[y_{cv_1}(t)]$ and the RDC method $[y(t)]$ with an oversampling ratio $F = 8.0$. Note that these results have the same shape. The theoretical maximum of the correlation is

$$\frac{3}{2}A^2 T_d = 1.5$$

Simulation results give a maximum value of correlation equal to 1.49999 when using the conventional method and 1.54536 using the RDC method.
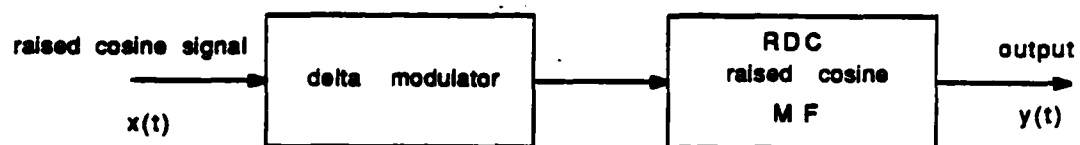
Fig. 6.12. Block diagram of the RDC matched filter that was simulated.
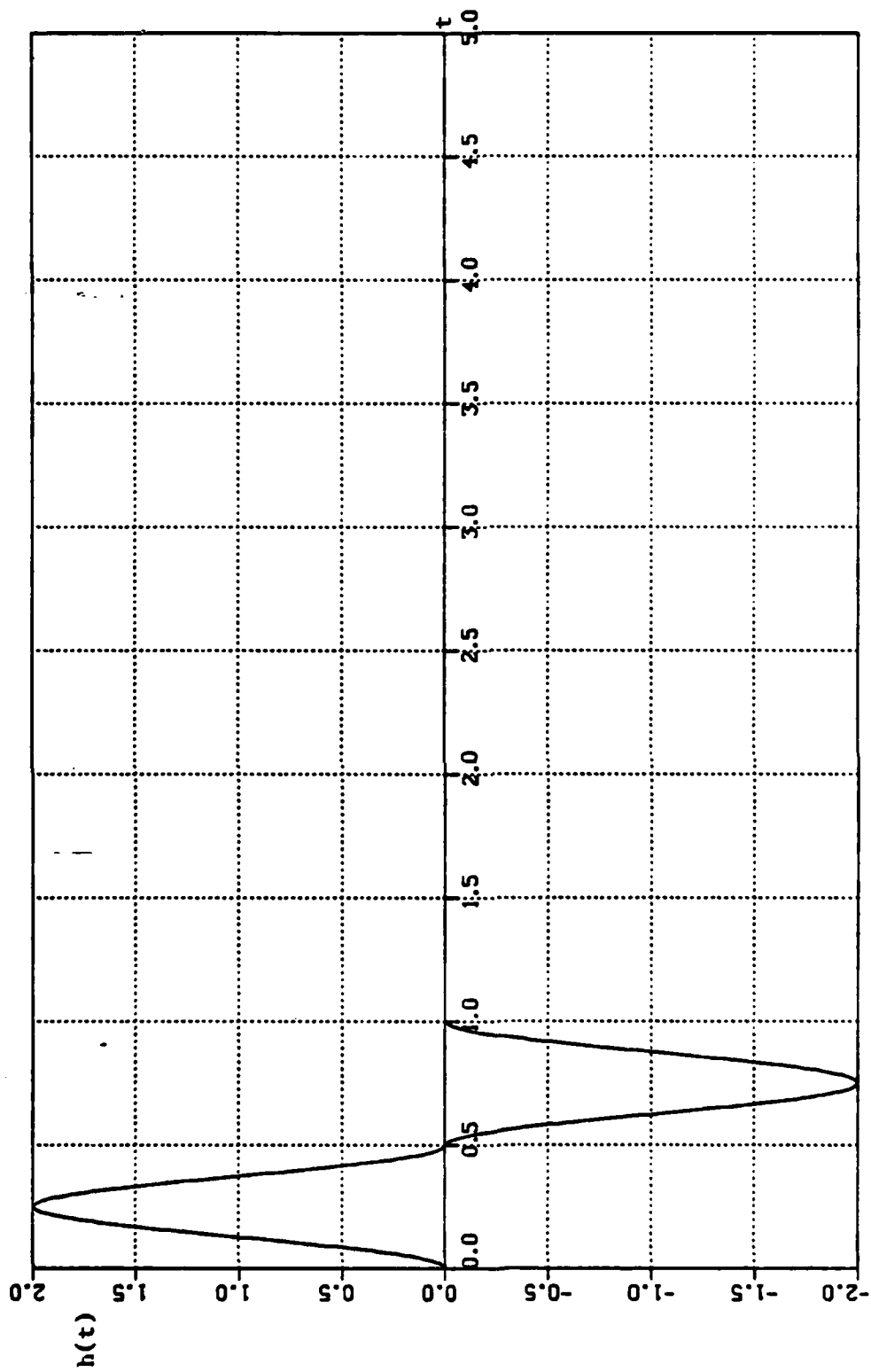
Fig. 6.13. Raised cosine MF impulse response.

93
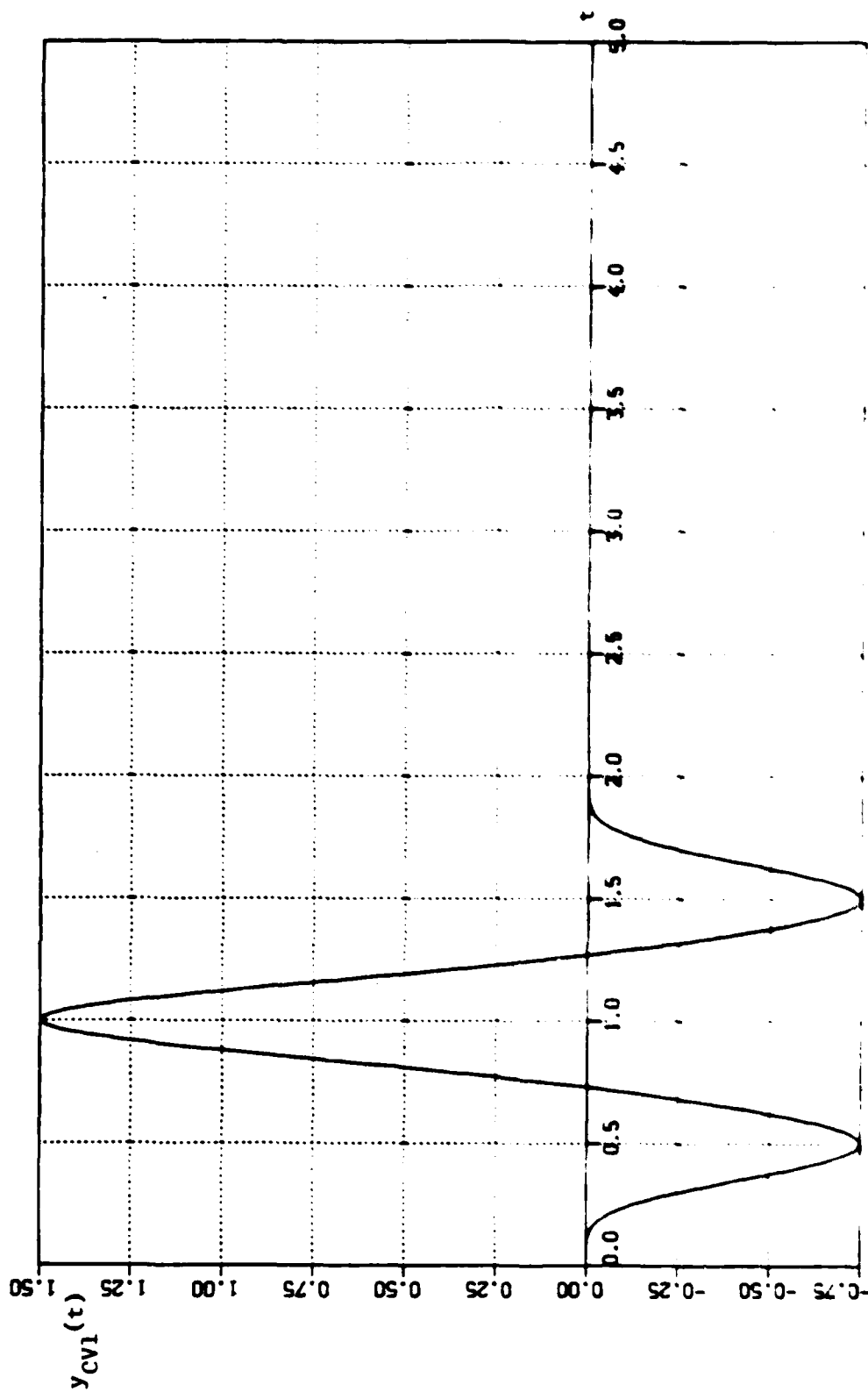
Fig 6.14 Output of the second coarse MF when conventional methods are employed

94
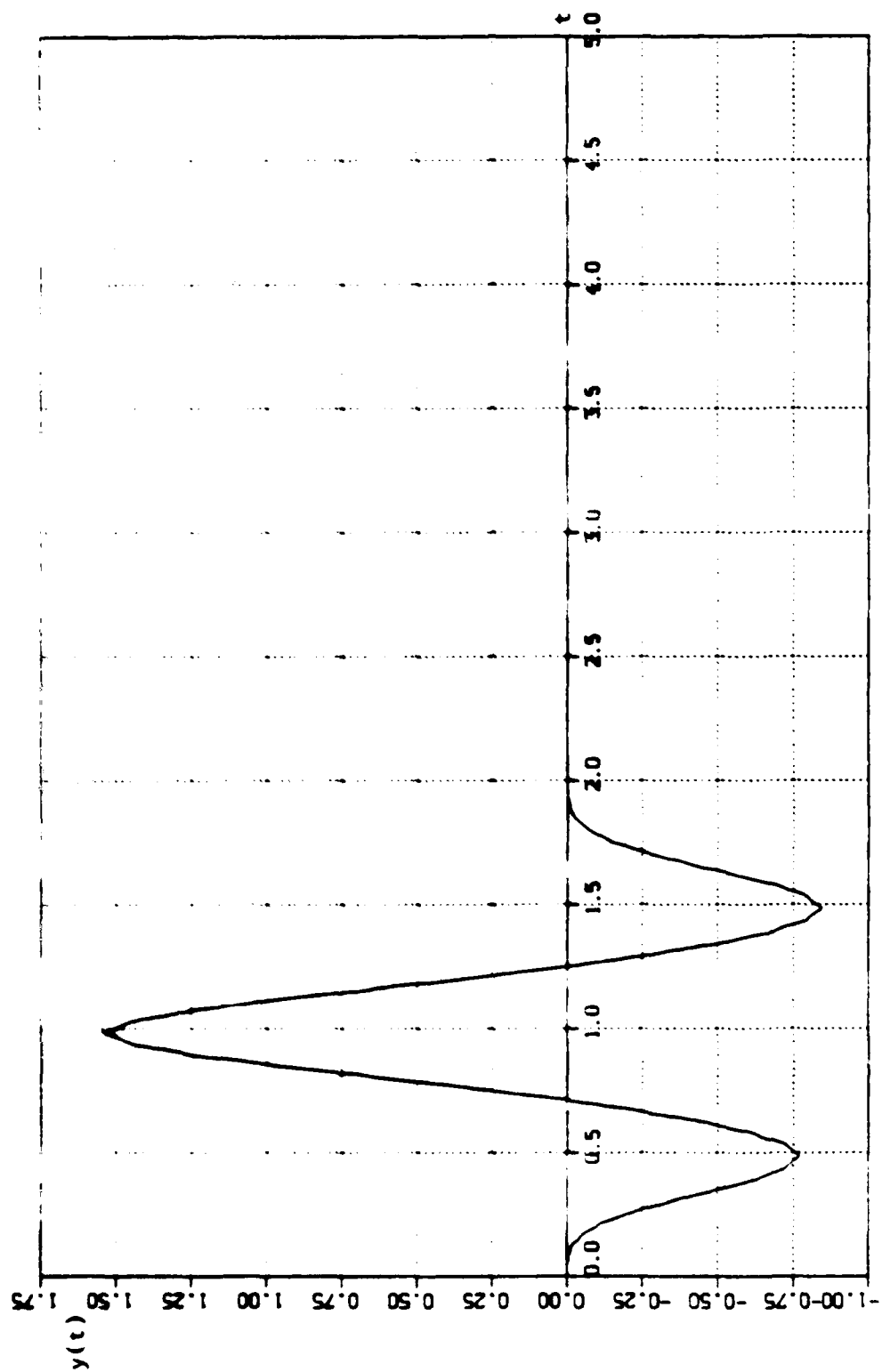
Fig. 6.15. Output of the raised cosine MF when the RDC method is employed when $F = 8$.
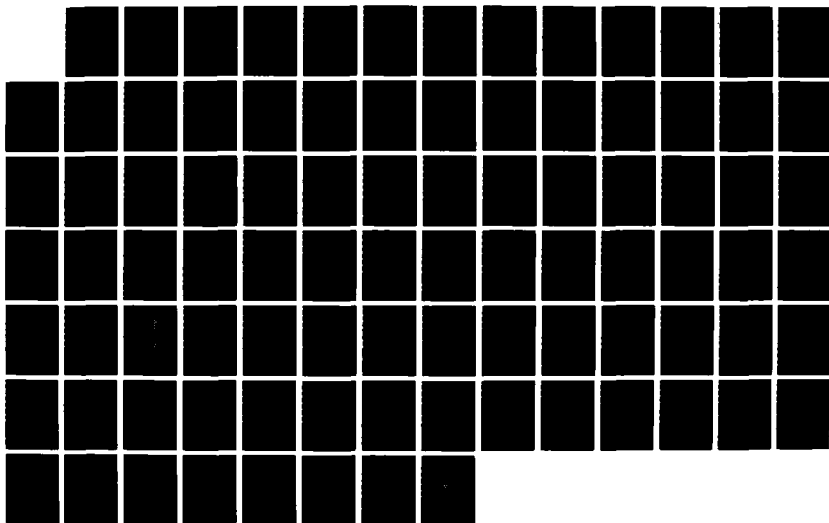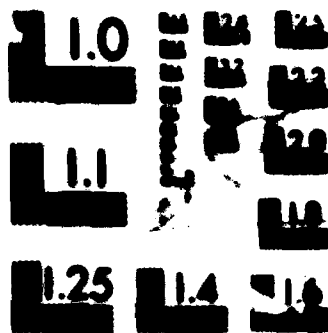
2. Chirp Matched Filter

The chirp or linearly frequency modulated signal has been popular in pulse compression applications in the past because it is possible to realize filters matched to the chirp signal using acoustic devices. The instantaneous frequency $f_i(t)$ as a function of time is given in Fig. 6.16 where $f_i(t) = f_1 + \frac{(f_2-f_1)}{T_d}t$ where T_d is the chirp pulse time duration. So, the chirp signal (voltage) is then equal

$$A \cdot \sin\left(w_1 t + \frac{(w_2 - w_1)}{2T_d}t^2\right)$$

A block diagram of the system simulated is shown as Fig. 6.17. Figures 6.18, 6.19, and 6.20 show the input signal $x(t)$, the output $y_{cv_1}(t)$ when we use a conventional method, and the output $y(t)$ of the RDC matched filter. System parameters used are $f_1 = 0.0$ Hz, $f_2 = 5.0$ Hz, $T_d = 4.0$ seconds, $A = 1.0$, $F = 4$, and $n_1 = 160$.

From Figures 6.19 and 6.20 we observe that the maximum of $y_{cv_1}(t)$ is 1.88820 and of $y(t)$ is 1.84669 where the theoretical maximum is 1.8879557. After the input expires the output is constant (non-zero). This dc offset is explained in Appendix D. From Equation (D.3) the dc offset is expected to be equal to $\pm\frac{\Delta x}{2}$ (total area under $h(t)$).

Increasing the oversampling ratio F to 32 decreases the dc offset. See Figures 6.20 and 6.21. This is so because when F increases, Δx decreases to keep the value of the slope-following capacity constant. And when Δx decreases, then from Equation (D.3), the dc offset decreases.

Note that in the case of the raised cosine matched filter, the result of correlation has no dc offset because the total area under $h(t)$ equals 0 (see Fig. 6.15).
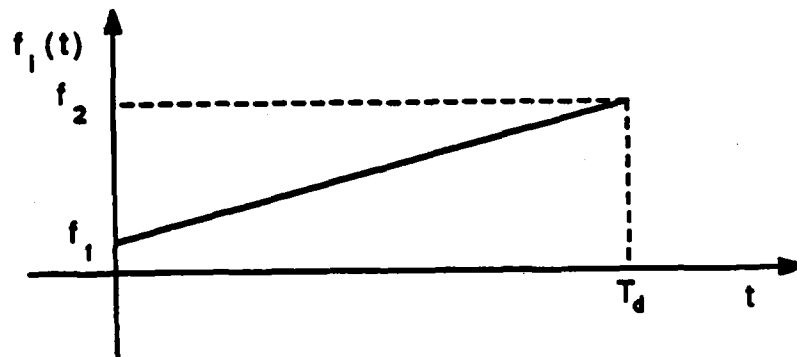
Fig. 6.16. The instantaneous frequency of the chirp signal.
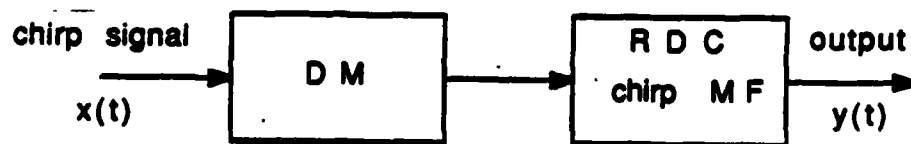
Fig. 6.17. A block diagram of the RDC chirp matched filter that was simulated.

97
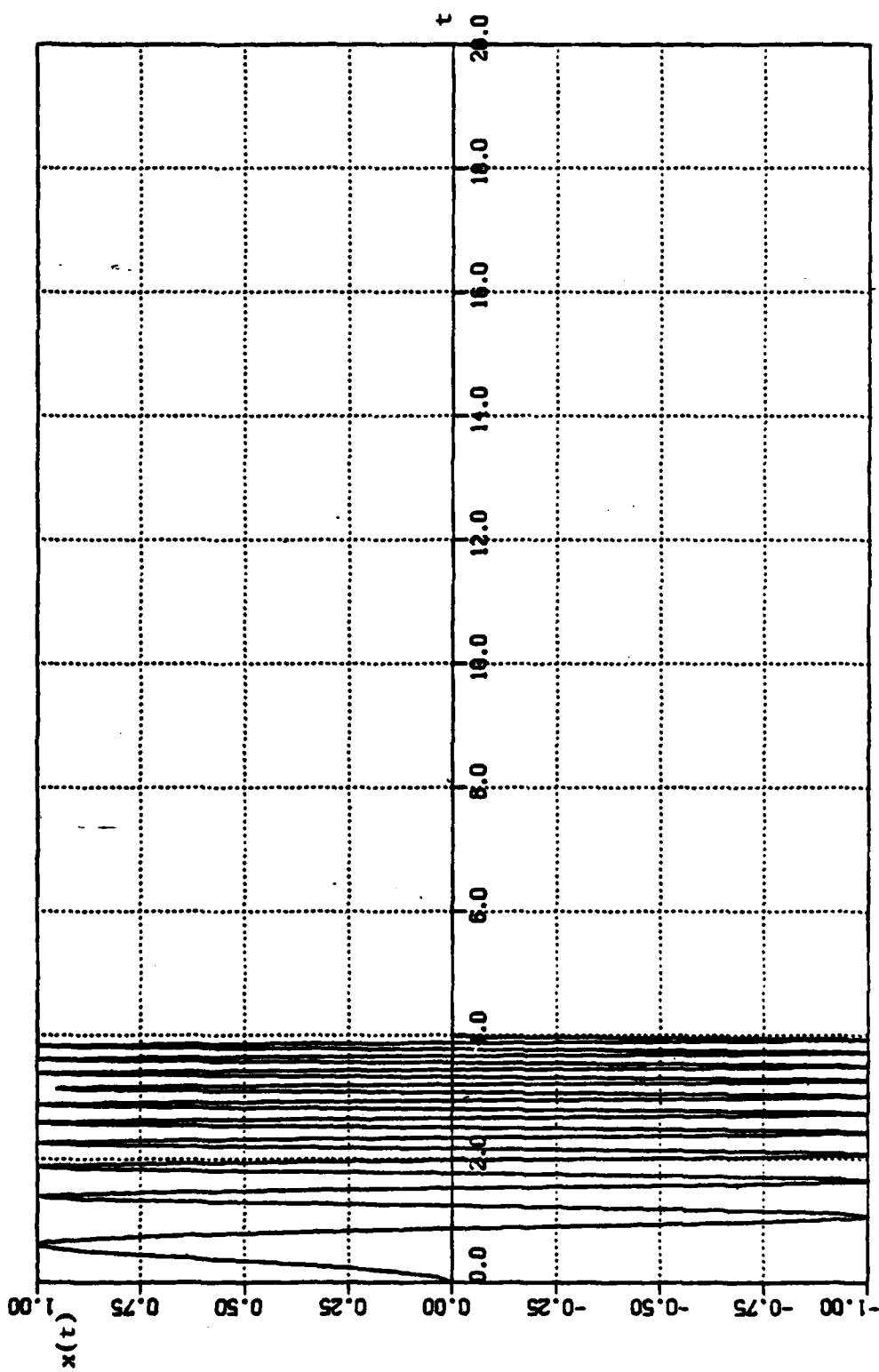
Fig. 6.18. The input chirp signal.
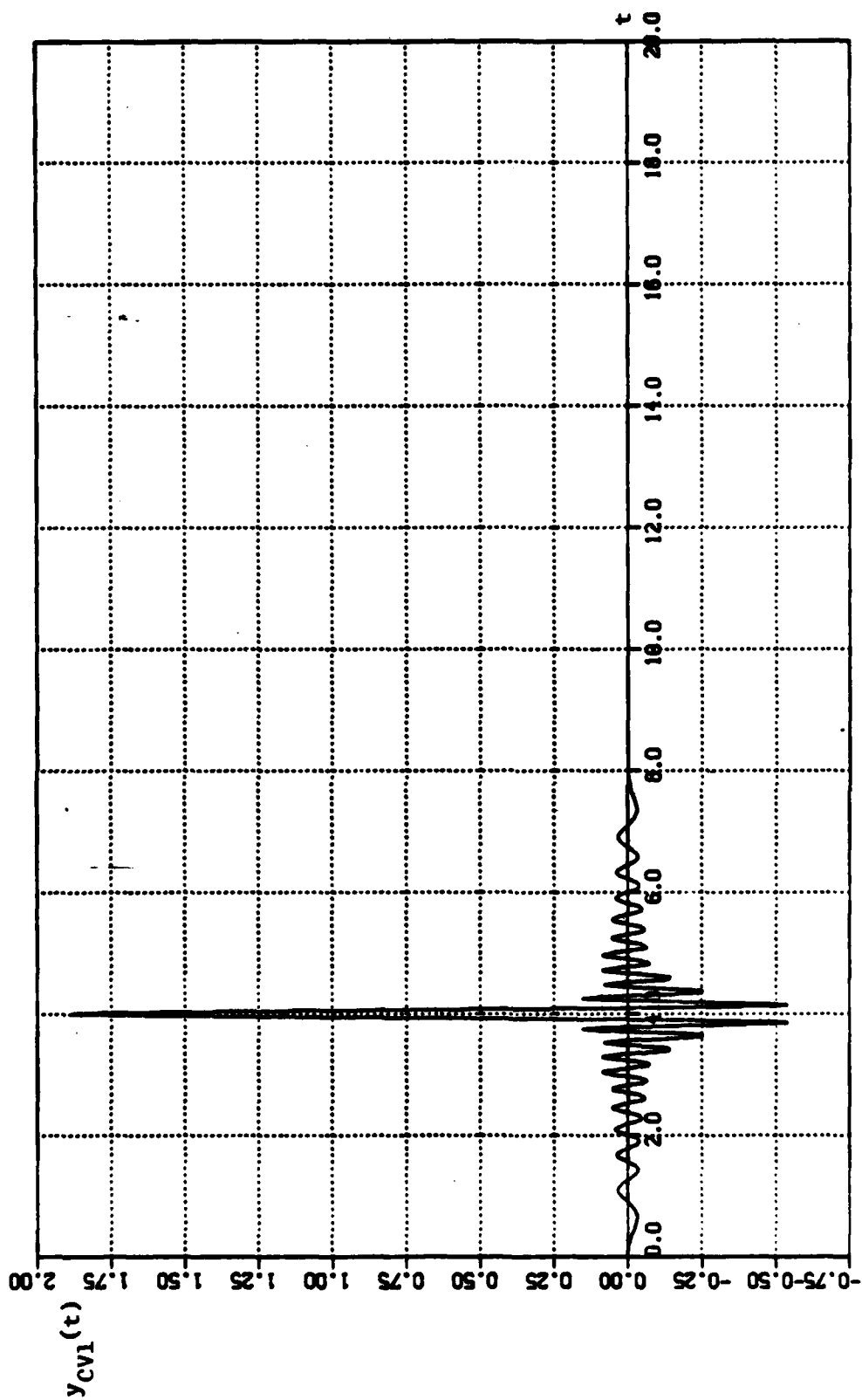
Fig. 6.19. The output of the chirp MF when using conventional methods.
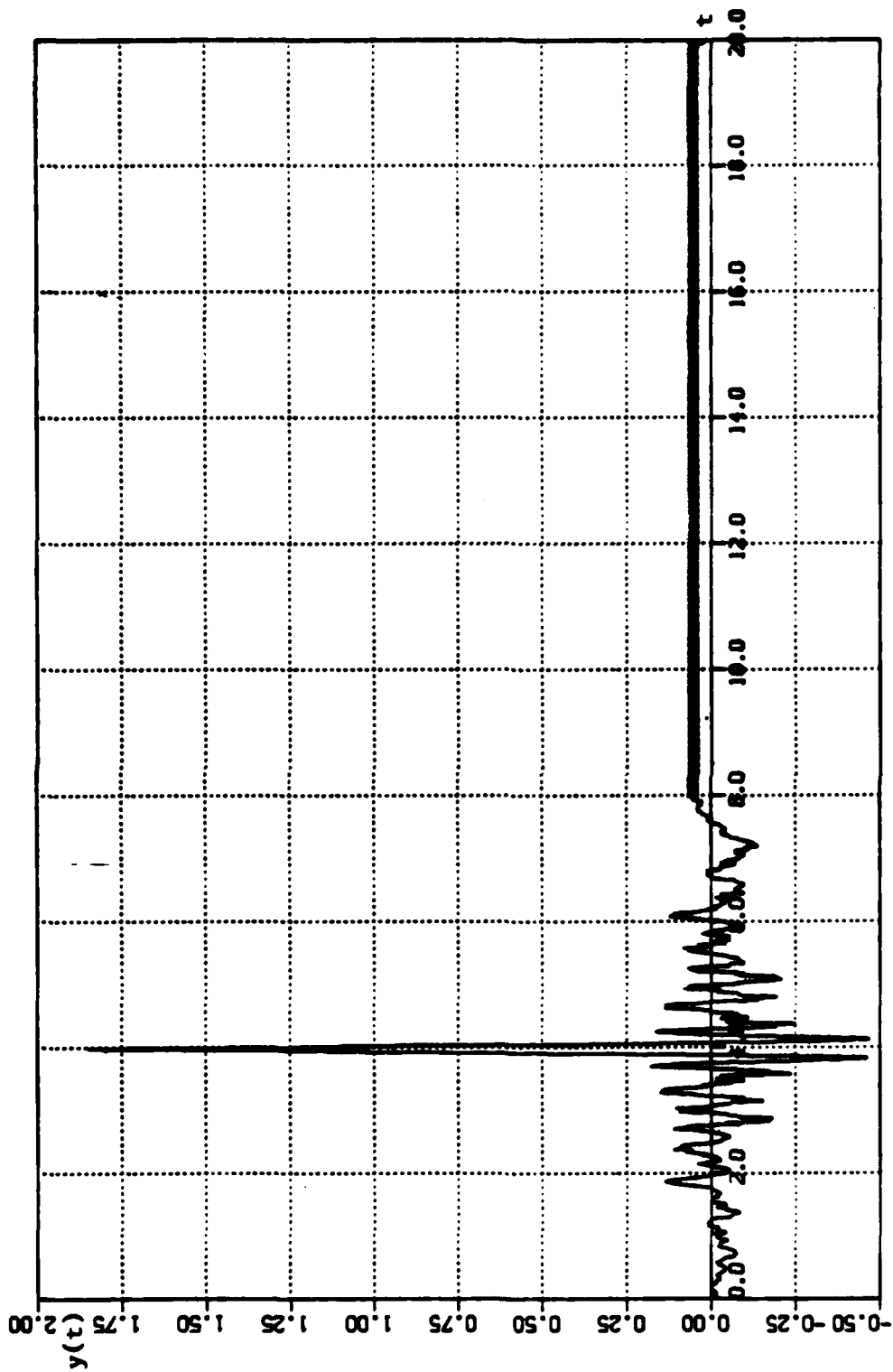
Fig. 6.20. The output of the chirp MF when using the RDC method for $F = 4$.
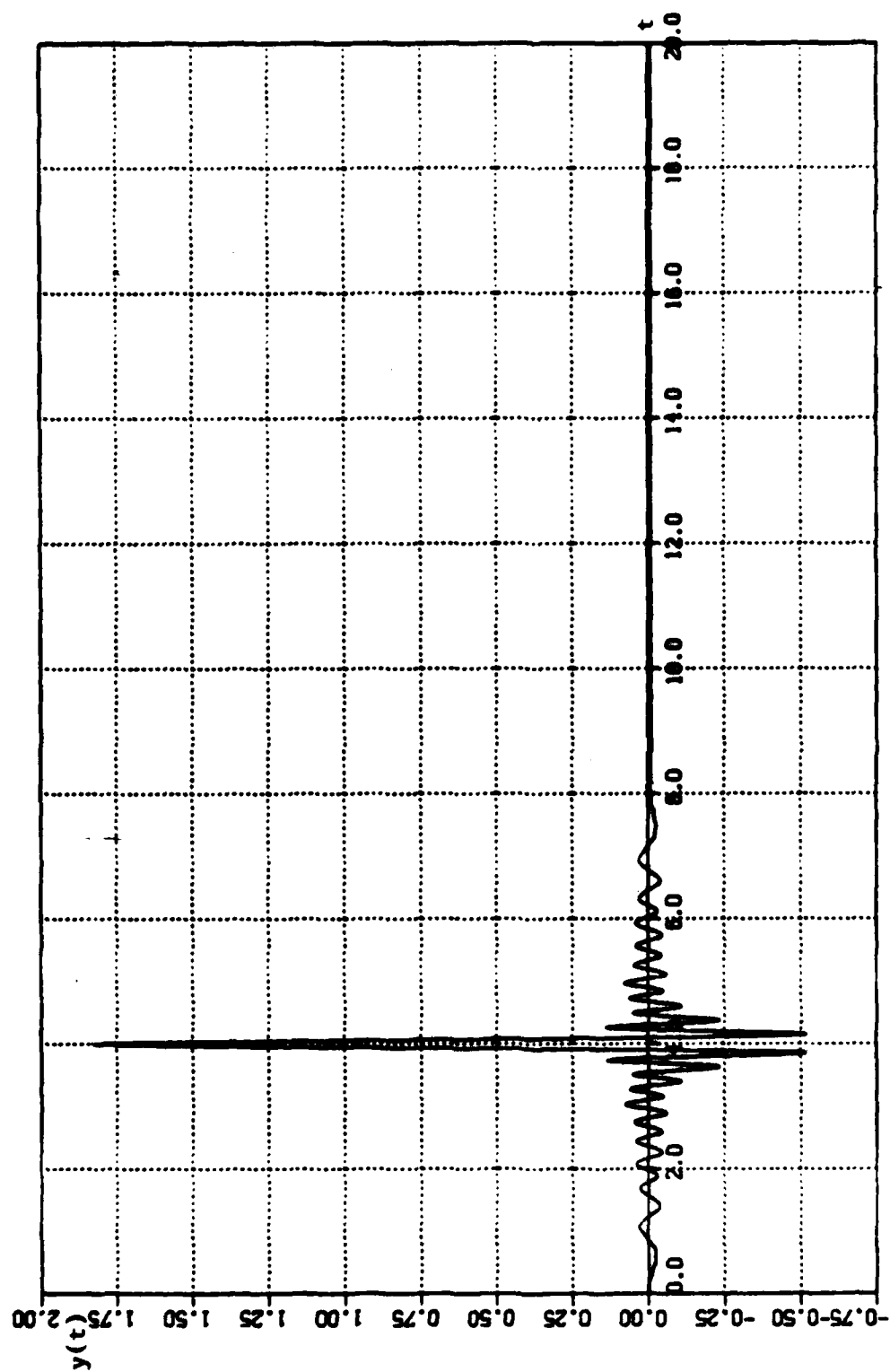
Fig. 6.21. The output of the chirp MF when using the RDC method for $F = 32$.

VII. NOISE PERFORMANCE OF THE RDC SYSTEM

A. INTRODUCTION

A DM converts an analog signal to a binary signal. This implies quantization. As a result of this and because of noise which may appear on the input, the output is subject to distortion. We identify two types of distortion and call their causes self noise and external noise. Self noise is caused by the delta modulation process. External noise is present at the input to the DM.

In this chapter we consider the effects on the output of the RDC system of first self noise and then external noise. Simulation results for lowpass filters and matched filters are presented. The idea of filtering on the basis of the slope of a voltage is introduced. Also included is the variable step size RDC method used to overcome a self noise effect on the representation of the impulse response of a filter.

Self noise is affected by the step size and sampling rate used in DM. Therefore, before considering self noise, we first establish an optimum step size.

B. OPTIMUM STEP SIZE

In this section we define and derive the optimum step size for use with the RDC method.

Let $x(t)$ be a zero-mean random input to the two level quantizer shown in Fig. 7.1. (In Fig. 3.1, the hard limiter, amplifier of gain G, and the flip flop can be replaced by a quantizer in the sense given in Fig. 7.1).

102
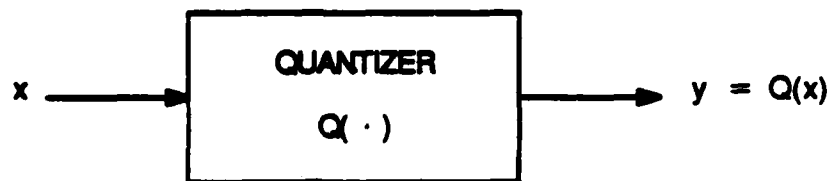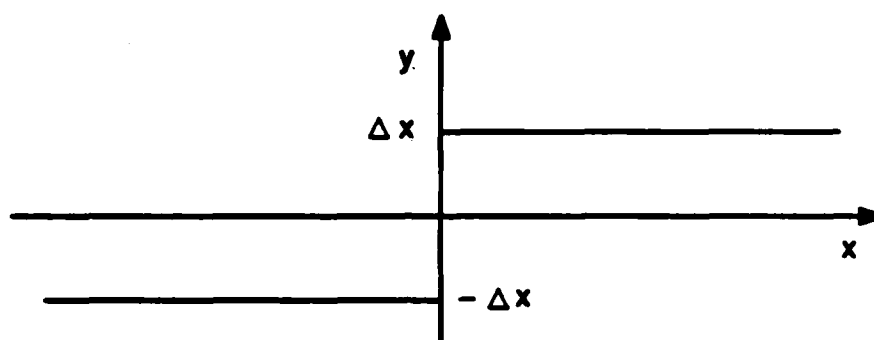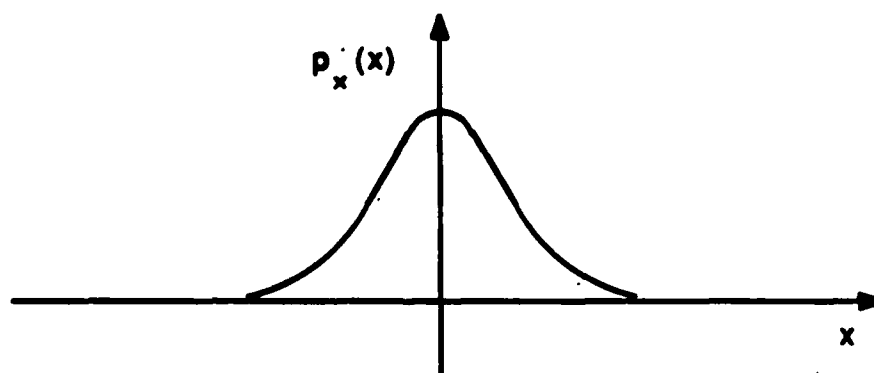
Fig. 7.1. Two level quantizer. (a) Block diagram. (b) Transfer characteristics.
(c) Example of input pdf.

103

Let $x(t)$ have a probability density function (pdf) $p_x(\cdot)$ and variance σ_x^2. (If the input $x(t)$ has a nonzero mean, we can subtract it from the input and add it back after quantization.). The quantizer output y is

$$y = Q(x) = \begin{cases} +\Delta x & \text{if } x \in (0, \infty) \\ -\Delta x & \text{if } x \in (-\infty, 0) \end{cases} \tag{7.1}$$

The quantization error $q_e = x - y$ is a random variable with pdf $p_q(\cdot)$ and variance σ_q^2, where

$$\sigma_q^2 = E\left[q_e^2\right] = \int_{-\infty}^{\infty} q_e^2 \cdot p_q(q) \cdot dq.$$

But,

$$q_e = x - Q(x) \text{ is a function of } x.$$

So,

$$\sigma_q^2 = E\left[q_e^2\right] = \int_{-\infty}^{\infty} (x - Q(x))^2 \cdot p_x(x) \cdot dx$$

$$= \int_{-\infty}^{0} (x - (-\Delta x))^2 \cdot p_x(x) \cdot dx + \int_{0}^{\infty} (x - \Delta x)^2 \cdot p_x(x) \cdot dx$$

$$= \int_{-\infty}^{\infty} x^2 \cdot p_x(x) \cdot dx + 2\Delta x \cdot \int_{-\infty}^{0} x \cdot p_x(x) \cdot dx$$

$$+ (\Delta x)^2 \cdot \int_{-\infty}^{\infty} p_x(x) \cdot dx - 2\Delta x \cdot \int_{0}^{\infty} x \cdot p_x(x) \cdot dx$$

If $p_x(x)$ is symmetric about zero, then

$$\sigma_q^2 = \sigma_x^2 - 4\Delta x \cdot \int_{0}^{\infty} x \cdot p_x(x) \cdot dx + (\Delta x)^2. \tag{7.2}$$

We define the optimum step size Δx_{opt} as that which minimizes σ_q^2. To find Δx_{opt}, form

$$\frac{d\sigma_q^2}{d(\Delta x)} = -4 \int_{0}^{\infty} x \cdot p_x(x) \cdot dx + 2\Delta x = 0$$

104

and solve for

$$\Delta x_{opt} = 2 \int_0^\infty x \cdot p_x(x) \cdot dx. \tag{7.3}$$

Using this value in Equation (7.2) gives

$$\sigma_q^2 = \sigma_x^2 - 2\Delta x \cdot \Delta x_{opt} + (\Delta x)^2 \text{ when } \Delta x \neq \Delta x_{opt} \tag{7.4}$$

and

$$min\left\{\sigma_q^2\right\} = \sigma_x^2 - (\Delta x_{opt})^2 \text{ when } \Delta x = \Delta x_{opt} \tag{7.5}$$

If we define a quantizer performance factor ϵ_q^2 as

$$\epsilon_q^2 = \frac{\sigma_q^2}{\sigma_x^2} \tag{7.6}$$

then,

$$min\left\{\epsilon_q^2\right\} = \frac{min\{\sigma_q^2\}}{\sigma_x^2} = 1 - \frac{(\Delta x_{opt})^2}{\sigma_x^2} \tag{7.7}$$

where the inverse of ϵ_q^2 can be considered as a signal to noise ratio.

Table 7.1 lists $\frac{\Delta x_{opt}}{\sigma_x}$ for different probability density functions and indicates the corresponding minimum value of ϵ_q^2. When the input is a sine wave, then the pdf is an arcsin function and $\frac{\Delta x_{opt}}{\sigma x} = 0.9$ (Jayant [31]). In all simulations this value of optimum step size was calculated for the applied voltage and then used to obtain the results.

The value of Δx_{opt} is that which minimizes the variance of the difference between the input and the output of the two-level quantizer. The two-level quantizer is part of a delta modulator. In fact the DM of Fig. 3.1 is formed by an integrator or accumulator in a feedback path from the output of a two-level quantizer to its input. We now have a new variable which is the gain g in the feedback loop. In the next section, we derive the optimum value of g.

105

TABLE 7.1. OPTIMUM STEP SIZE AND MINIMUM VALUE

OF THE QUANTIZER PERFORMANCE FACTOR FOR

DIFFERENT INPUT PDF'S.

pdf	$\frac{\Delta_{opt}}{\sigma_x}$	$min\{\epsilon_q^2\}$
Uniform	$\frac{\sqrt{3}}{2}$	0.25
Gaussian	$\sqrt{\frac{2}{\pi}}$	0.363
Laplace	$\frac{1}{\sqrt{2}}$	0.5
Gamma	$\frac{1}{\sqrt{3}}$	0.667

1. Optimum Gain of DM Feedback Loop

In Fig. 7.2, assume that the accumulator has an inherent 1 bit delay; then,

$$x_s(n \cdot T_s) = g \cdot x([n-1] \cdot T_s).$$

As a result of this

$$x_\Delta(n \cdot T_s) = x(n \cdot T_s) - x_s(n \cdot T_s) = x(n \cdot T_s) - g \cdot x([n-1] \cdot T_s)$$

$$\overline{(x_\Delta(n \cdot T_s))^2} = \overline{(x(n \cdot T_s) - g \cdot x([n-1] \cdot T_s))^2} = \sigma_\Delta^2.$$

Therefore,

$$\sigma_\Delta^2 = \sigma_x^2 + g^2 \cdot \sigma_x^2 - 2g \cdot \rho_1 \cdot \sigma_x^2 \tag{7.8}$$

where σ_Δ^2 is the variance of the input signal to the quantizer (hard limiter) and ρ_1 is the value of the variance normalized autocorrelation function $R_{xx}(\tau)$ of $x(t)$ for $\tau = T_s$.

Setting $\frac{\partial \sigma_\Delta^2}{\partial g} = 0$ yields the optimum value of g and $min\{\sigma_\Delta^2\}$ the minimum error variance.

$$\frac{\partial \sigma_\Delta^2}{\partial g} = (2g - 2\rho_1) \cdot \sigma_x^2 = 0$$
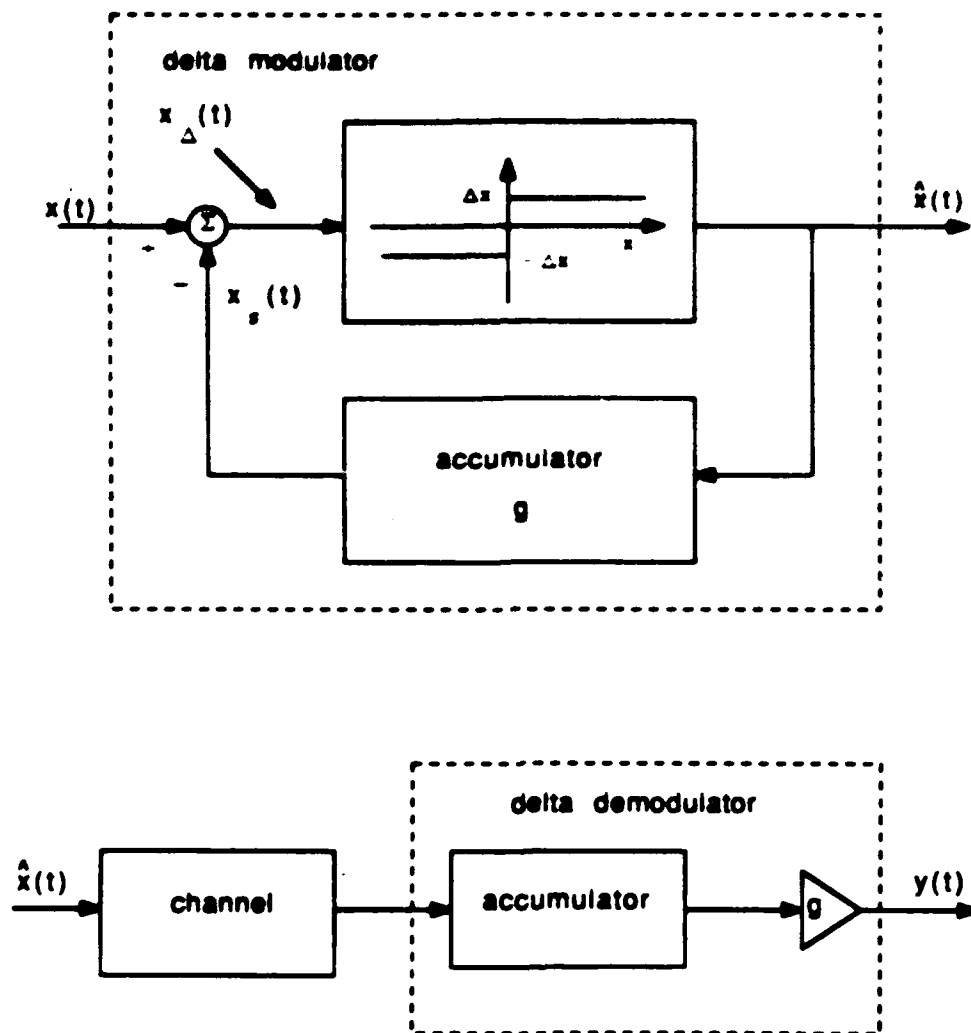
Fig. 7.2. Block diagram of the delta modulator and demodulator.

Therefore,

$$g_{opt} = \rho_1 \text{ and } min\{\sigma_\Delta^2\} = (1 - \rho_1^2)\sigma_x^2 \tag{7.9}$$

We see from Equation 7.9 that the error variance is less than the input signal variance σ_x^2 for all non-zero values of ρ_1. We call $\frac{\sigma_x^2}{\sigma_\Delta^2}$ a prediction gain G_P.

The maximum value of $G_P = max\{G_P\} = \dfrac{\sigma_x^2}{min\{\sigma_\Delta^2\}} = (1 - \rho_1^2)^{-1}$ \qquad (7.10)

Processors of speech and images are often based on long-term autocorrelation functions. In general, for speech ρ_1 has a value close to 0.9 (Jayant [31]). In sub-optimal design, ρ_1 is taken as 1 and so $g = 1$.

The preceding consideration of a two-level quantizer and feedback gain can now be applied to determine the optimum step size for DM analog-to-binary conversion.

2. <u>Optimum Step Size for Delta Modulation</u>

From Fig. 7.2 we can define for a DM signal the reconstruction error $r(n \cdot T_s)$ as follows

$$r(n \cdot T_s) = x(n \cdot T_s) - y(n \cdot T_s)$$

where $y(n \cdot T_s)$ is the output of the delta demodulator which is equal to $x_s(n \cdot T_s)$. We can prove that the reconstruction error has a minimum variance

$$min\left\{\sigma_r^2\right\} = min\left\{\sigma_q^2\right\} = \frac{min\left\{\in_q^2\right\} \cdot (1 - \rho_1^2)}{\left(1 - \rho_1^2 \cdot min\left\{\in_q^2\right\}\right)}\sigma_x^2 \tag{7.11}$$

where we note that the minimization in Equation (7.11) is with respect to Δx and that this result holds for $g \approx \rho_1$ (Jayant [31]).

From Equation (7.7) we can say that

$$\Delta x_{opt} = \left(\sqrt{1 - min\left\{\in_q^2\right\}}\right) \cdot \sigma_\Delta$$

$$= \Theta \cdot \sigma_\Delta \tag{7.12}$$

108

where Θ is a constant which depends on the pdf of the input $x(t)$. Also, from Equations 7.7 and 7.11

$$\sigma_\Delta^2 = \frac{(1 - \rho_1^2)}{(1 - \rho_1^2 \cdot min\{\epsilon_q^2\})}\sigma_x^2. \tag{7.13}$$

So, from Equations (7.12) and (7.13)

$$\Delta x_{opt} = \Theta \cdot \sqrt{\frac{(1 - \rho_1^2)}{(1 - \rho_1^2 \cdot min\{\epsilon_q^2\})}}\ \sigma_x \tag{7.14}$$

where $min\{\epsilon_q^2\} < 1$, and can be found using Equations (7.3) and (7.7) or Table 7.1 for the pdf of interest.

If $\rho_1 \rightarrow 1$ (which is the case for speech and video signals for sampling frequencies higher than the Nyquist frequency), Equation (7.14) can be rewritten as follows

$$\Delta x_{opt} = \Theta \cdot \sqrt{\frac{(1 - \rho_1^2)}{(1 - min\{\epsilon_q^2\})}}\ \sigma_x.$$

Then using Equation (7.12),

$$\Delta x_{opt} = \sqrt{1 - \rho_1^2}\ \sigma_x. \tag{7.15}$$

For pdf's of many common input signals,

$$max\{G_P\} = \frac{1}{1 - \rho_1^2} \propto F^2$$

(Jayant [31]) where F is the DM oversampling ratio defined in Section V.C. Let

$$max\{G_P\} = C_{ps}^2 F^2 = \frac{1}{1 - \rho_1^2} \tag{7.16}$$

where C_{ps} is a constant. From Equations (7.15) and (7.16)

$$\Delta x_{opt} = \frac{\sigma_x}{C_{ps} \cdot F} \tag{7.17}$$

109

For example, for a sinusoid of frequency f_1 and amplitude A which is sampled at frequency f_s

$$x(n \cdot T_s) = A \cdot \sin(w_1 \cdot n \cdot T_s)$$

Therefore,

$$R_{xx}(T_s) = A^2 \overline{\sin(w_1 \cdot n \cdot T_s) \cdot \sin(w_1 \cdot [n+1] \cdot T_s)}$$
$$= \frac{A^2}{2} \overline{\cos(w_1 \cdot T_s) - \cos(w_1(2n+1) \cdot T_s)}$$
$$= \frac{A^2}{2} \cos(w_1 \cdot T_s)$$
$$= \frac{A^2}{2} \cos\left(\frac{2\pi f_1}{f_s}\right) = \frac{A^2}{2} \cos\left(\frac{\pi}{F}\right) \tag{7.18}$$

and

$$R_{xx}(0) = \frac{A^2}{2}. \tag{7.19}$$

So,

$$\rho_1 = \frac{R_{xx}(T_s)}{R_{xx}(0)} = \cos\left(\frac{\pi}{F}\right).$$

But

$$\cos x = 1 - \frac{x^2}{2} + \frac{x^4}{4!} - \dots$$

Therefore,

$$\rho_1 = \cos\left(\frac{\pi}{F}\right) \approx 1 - \frac{1}{2}\left(\frac{\pi}{F}\right)^2 \quad \text{where } F >> 1.$$

So,

$$(1 - \rho_1^2) = \left(\frac{\pi}{F}\right)^2 - \frac{1}{4}\left(\frac{\pi}{F}\right)^4 \approx \left(\frac{\pi}{F}\right)^2.$$

As a result of this

$$max\{G_P\} = \left(1 - \rho_1^2\right)^{-1} = \frac{F^2}{\pi^2}.$$

From Equation 7.16

$$C_{ps} = \frac{1}{\pi}. \tag{7.20}$$

110

From Equations (7.17) and (7.20)

$$\Delta x_{opt} = \frac{\pi}{F}\sigma_x. \tag{7.21}$$

But

$$\sigma_x = \frac{A}{\sqrt{2}}.$$

Therefore,

$$\Delta x_{opt} = \frac{\sqrt{2}\,\pi \cdot A \cdot f_1}{f_s} \approx \frac{4.44\,A \cdot f_1}{f_s} = \frac{2.22\,A}{F}. \tag{7.22}$$

Having established optimum values of step size and gain for a DM, we now can use these results to determine values of self noise in the ADC process using DM.

C. SELF NOISE

Chapter Three identifies two types of self noise: granular (quantization) noise and slope overload noise.

1. Granular Noise Power

If $|x(t)|$ is less than the step size as shown in Fig. 7.3, the granular noise is given by

$$\epsilon_g = x(t) - x_s(t)$$

Generally, we have granular noise for all $x(t)$ because of quantization. To precisely define ϵ_g, let us assume that the analog input to the DM modulator $x(t)$ and the output of the DM receiver are given in Fig. 7.4. Assume also that the step size Δx and the sampling interval T_s are infinitesimal, but with their ratio the same as the actual system. The output of the DM receiver will then be the smooth function $x_{ss}(t)$ shown in Fig. 7.5.

111

We define slope overload noise ϵ_{so} as the difference between $x(t)$ and $x_{so}(t)$; the remaining distortion $(x_{so}(t) - x_s(t))$, is granular noise.

Granular noise ϵ_g is essentially uncorrelated with the signal (Peebles [32] and Protonotarios [33]) and is approximately uniformly distributed (Goodman [29] and Protonotarios [33]), over the interval $(-\Delta x, \Delta x)$. See Fig. 7.6. Therefore,

$$E(\epsilon_g) = \int_{-\Delta x}^{\Delta x} \epsilon_g \cdot \left(\frac{1}{2\Delta x}\right) \cdot d\epsilon_g = 0,$$

and the granular noise variance is given by

$$\sigma_g^2 = \int_{-\Delta x}^{\Delta x} (\epsilon_g)^2 \left(\frac{1}{2\Delta x}\right) d\epsilon_g = \frac{(\Delta x)^2}{3}. \tag{7.23}$$

So, the granular noise power increases as the square of the step size.

2. Slope Overload Noise

In general the slope overload noise is correlated with the message (Peebles [32]). A number of analyses to find the slope overload noise power are introduced by Protonotarios [33], Zetterberg [34], Rice (with O'Neal) [35], and Abate [36]. All these results do not either individually or collectively pertain to all slope-following capacities and input spectra. Greenstein [30] found an expression for slope overload noise power that is accurate for all slope-following capacities and input spectra of possible interest for linear delta modulators having Gaussian random inputs.

Let us define the slope overload factor

$$s \triangleq \frac{x_o'}{\left(\frac{dx}{dt}\right)_{r.m.s}} \tag{7.24}$$

where x_o' is the slope-following capacity. A highly accurate approximation to the slope overload noise power σ_{so}^2 for linear delta modulators having Gaussian inputs and for all spectra, that was given by Greenstein [30], is as follows
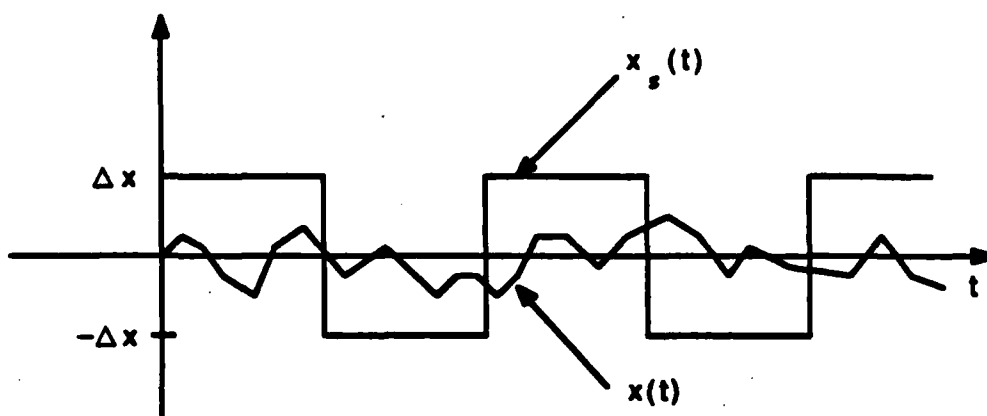
Fig. 7.3. Illustration of the case where $|x(t)| < \Delta x$.

113
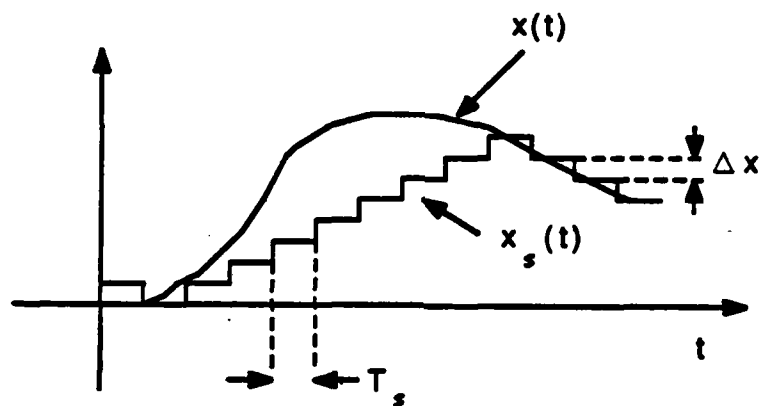
Fig. 7.4 The input signal and its staircase representation.
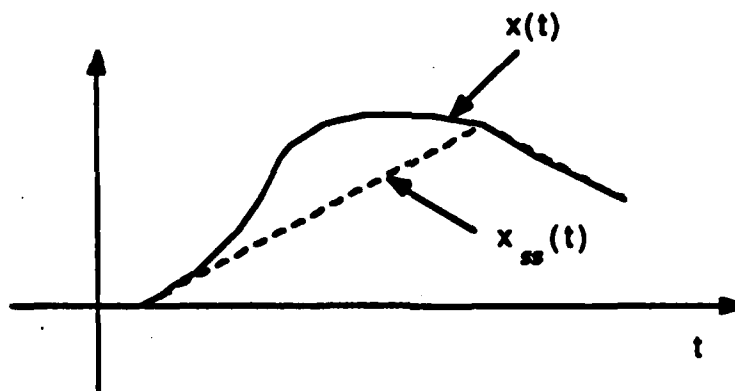
Fig. 7.5. The input signal and the smooth staircase representation.

114
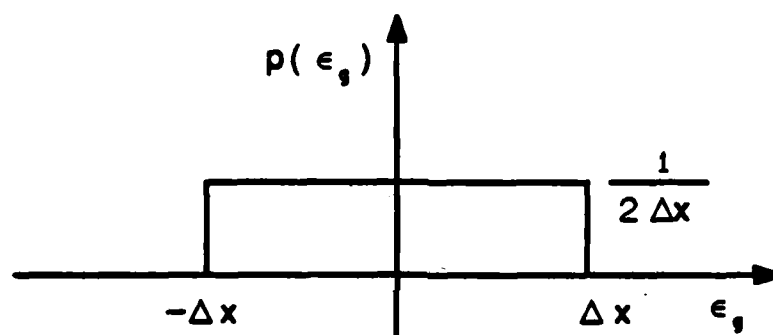
Fig. 7.6. Probability density function of granular noise.

$$\sigma_{so}^2 = \sigma_x^2 \left[1 + 2.753s + 2.952s^2\right] \exp(-0.341s^2)$$

$$\cdot \exp\left\{(a_1 - 2.753)s + a_2 \left[\exp(a_3 s + a_4 s^2) - 1\right]\right\}; 0 < s < 4.0 \quad (7.25)$$

where the expression given by Equation (7.25) predicts σ_{so}^2 with an accuracy of 1 dB or better for s between 4.0 and 6.5. For $s \geq 6.5$, σ_{so}^2 is at least 119 dB below σ_x^2. The variables a_1, a_2, a_3, and a_4 are determined by the spectrum of the input (Greenstein [30]). For example, for bandlimited white noise

$$a_1 = -0.036, a_2 = 0.37, a_3 = -3.83, \text{ and } a_4 = -5.9.$$

(See Fig. 7.7.) From this section and section VII.C.1 we can draw an error variance curve for DM (See Fig. 7.8.), where

$$\sigma_r^2 = \sigma_{so}^2 + \sigma_g^2 \quad (7.26)$$

Note that as Δx falls below the value Δx_{opt}, the distortion increases more rapidly than when Δx takes on values greater than Δx_{opt}. We say that the delta modulator "fits" the signal if $\Delta x = \Delta x_{opt}$.

3. **Signal to Noise Ratio Improvement (SNRI) Due to Delta Modulation**

Assume that we have chosen the step size to be equal to Δx_{opt} where we have as input of the RDC system the signal alone. Then,

$$\sigma_{r_s}^2 = \sigma_{so_s}^2 + \sigma_{g_s}^2 \approx \sigma_{g_s}^2 = \frac{(\Delta x_{opt})^2}{3} \quad (7.27)$$

where $\sigma_{r_s}^2, \sigma_{so_s}^2$, and $\sigma_{g_s}^2$ are the total quantization noise power, the slope overload noise power and the granular noise power respectively.

116
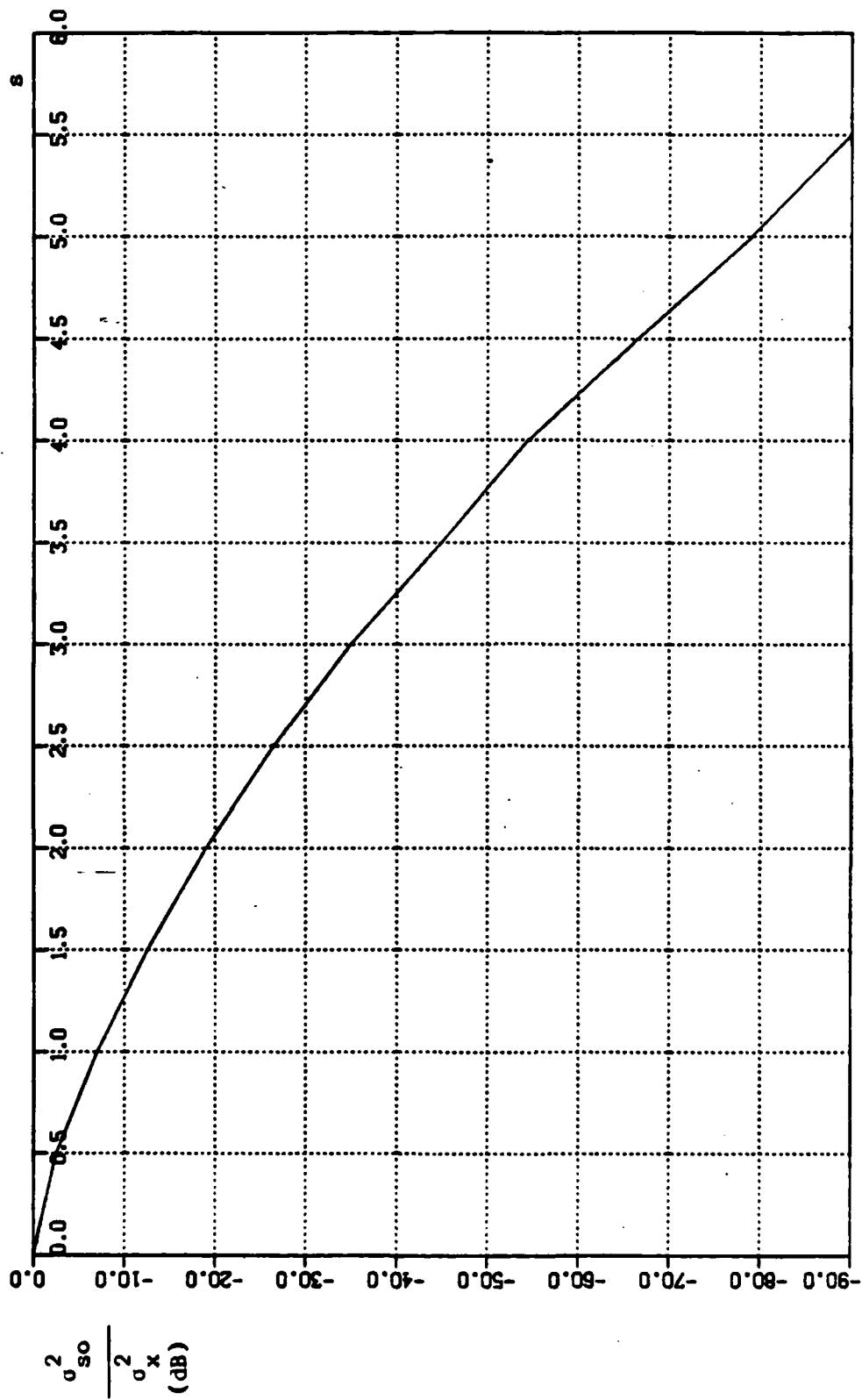
Fig. 7.7. Normalized slope overload noise as a function of, s.

117
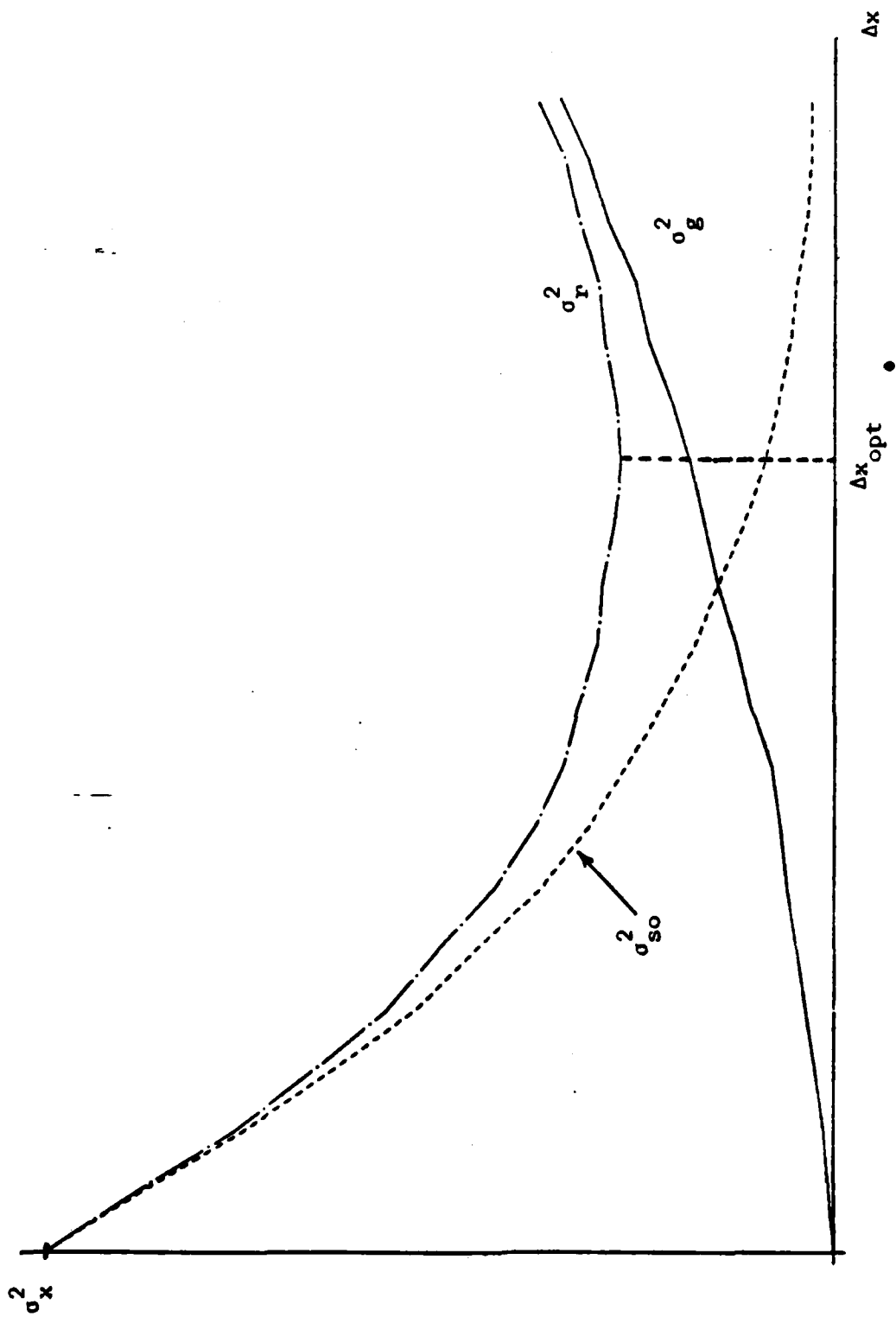
Fig. 7.8. Error variance curve for DM.

If we have as input noise alone (which has power σ_n^2) and the step size is Δx_{opt} then

$$\sigma_{r_n}^2 = \sigma_{so_n}^2 + \sigma_{g_n}^2 \qquad (7.28)$$

where $\sigma_{r_n}^2, \sigma_{so_n}^2$, and $\sigma_{g_n}^2$ are the total quantization noise power, the slope overload noise power, and the granular noise power respectively.

Neglecting the granular noise, which is practically very small compared to the slope overload noise, the effective noise to the RDC system will be decreased by $\sigma_{so_n}^2$. That is,

$$\text{the effective input noise power } = \sigma_n^2 - \sigma_{so_n}^2 = \sigma_n^2(1 - A) \qquad (7.29)$$

where

$$A = \frac{\sigma_{so_n}^2}{\sigma_n^2} < 1. \qquad (7.30)$$

See Fig. 7.9. From Equations (7.25), (7.29), and (7.30) we can find the input effective noise power when we have noise as input.

When we have as input to the RDC system the signal as well as noise, the root mean square value of the slope of the input will increase. From Equation (7.24) as $\left(\frac{dx}{dt}\right)_{rm}$ increases (where Δx and T_s are not changed), s decreases. From Fig. 7.7 as s decreases, the slope overload noise power, in this case $\sigma_{so_{x+n}}^2$, increases. Neglecting the granular noise, the output power of the RDC system (neglecting any SNRI due to matched filtering) is

$$\sigma_x^2 + \sigma_n^2 - \sigma_{so_{x+n}}^2$$

Therefore,

$$\frac{(SNR)_o}{(SNR)_i} = \left(\frac{\sigma_x^2}{\sigma_n^2 - \sigma_{so_{x+n}}^2}\right)\left(\frac{\sigma_n^2}{\sigma_x^2}\right)$$
$$= \frac{1}{1 - D} \qquad (7.31)$$

119
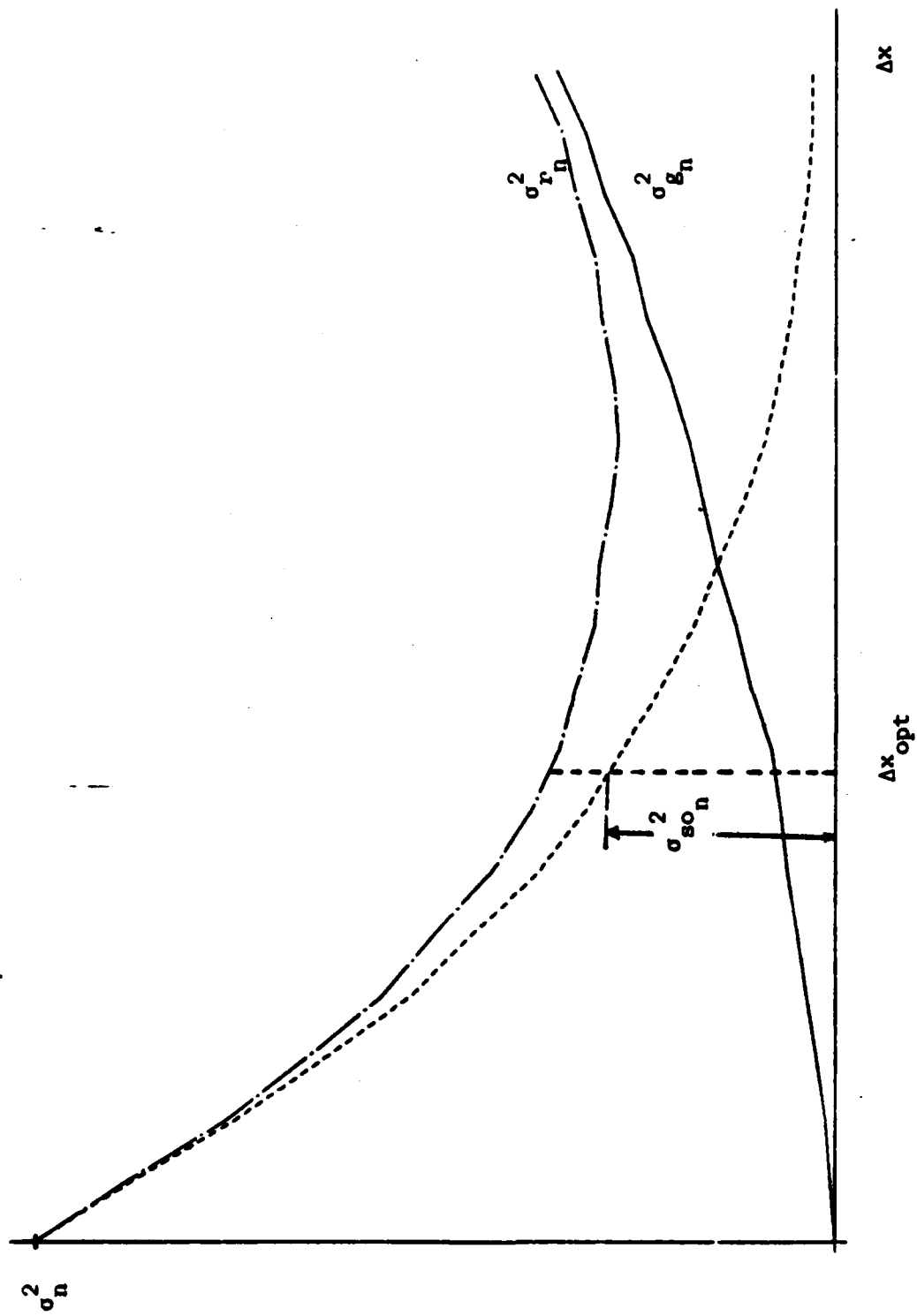
Fig. 7.9. Error variance when the input does not fit the delta modulator.

120

where

$$D = \frac{\sigma^2_{so_s+n}}{\sigma^2_n} < 1$$

So, the signal to noise ratio improvement due to using delta modulation = $\frac{1}{1-D} > 1$.

D. THE ERROR IN CONVOLUTION WHEN THE OUTPUT BITS OF THE DELTA MODULATOR ARE IN ERROR

The DM is an ADC. The bit stream output can be transmitted as a baseband signal (twisted pair, coaxial cable, or fiber) or as a bandpass signal (modulated carrier). Errors may occur in transmission.

A known signal at the input to a DM creates a known bit stream. Noise added to the signal creates a different bit stream. The difference can be considered as bits in error. The relationship between the noise characteristics and the bits in error is not generally known.

In this section, we consider the effect on the output of the RDC system of bits in error. Because there are two feedback circuits in the RDC system, bits in error on the input affect future values of the output.

From first principles of delta convolution (Appendix A), the error in the convolution due to one bit in error is given by

$$\epsilon_{conv} = \pm 2C_1 \cdot \left\{ \sum_{i=0}^{n_1-1} h_i \cdot (n_1 - i) \right\} \quad (7.32)$$
$$= \pm \epsilon_1$$

where ϵ_1 is a constant for a given $h(t)$ and T_s.

Assume that we have two bits in error. Then the bits in error can be as follows. Both can be $+1$; both can be -1; or they can be of opposite sign. There are

$\binom{2}{2}$ combinations of two 1's

121

$\binom{2}{1}$ combinations of a single 1 and a single -1,

and

$\binom{2}{0}$ combinations of two -1's.

where $\binom{m}{n}$ is the binomial coefficient of m things taken n at a time.

Therefore the error in convolution will be

$$2\epsilon_1 \text{ with probability } \frac{\binom{2}{2}}{2^2} = \frac{1}{4}$$

$$\bullet \quad 0 \text{ with probability } \frac{\binom{2}{1}}{2^2} = \frac{2}{4}$$

and

$$-2\epsilon_1 \text{ with probability } \frac{\binom{2}{0}}{2^2} = \frac{1}{4}.$$

Generally, if we have m bits in error and m is even, the error in convolution is given by

$$
\begin{array}{lll}
m \cdot \epsilon_1 & \text{with probability} & \frac{\binom{m}{m}}{2^m} \\[2ex]
(m-2) \cdot \epsilon_1 & \text{with probability} & \frac{\binom{m}{m-1}}{2^m} \\[2ex]
(m-4) \cdot \epsilon_1 & \text{with probability} & \frac{\binom{m}{m-2}}{2^m} \\[1ex]
\vdots & & \\[1ex]
2\epsilon_1 & \text{with probability} & \frac{\binom{m}{m-\left(\frac{m-2}{2}\right)}}{2^m} \\[2ex]
0 & \text{with probability} & \frac{\binom{m}{m-\frac{m}{2}}}{2^m} \\[1ex]
\vdots & & \\[1ex]
(m-k)\epsilon_1 & \text{with probability} & \frac{\binom{m}{m-\frac{k}{2}}}{2^m} \\[1ex]
\vdots & & \\[1ex]
\text{and } -m \cdot \epsilon_1 & \text{with probability} & \frac{\binom{m}{0}}{2^m}
\end{array}
$$

122

where k is even number. See Fig. 7.10.

So,

$$\overline{\epsilon_{conv}} = 0$$

and

$$\overline{\epsilon_{conv}^2} = \frac{2}{2^m} \sum_{k=0}^{m-2} ((m-k) \cdot \epsilon_1)^2 \binom{m}{m-\frac{k}{2}} \tag{7.33}$$

where m and k are even numbers.

In the same way, if we have m bits in error, where m is an odd number, the error in convolution is given by

$$(m-k) \cdot \epsilon_1 \text{ with probability } \frac{\binom{m}{m-\frac{k}{2}}}{2^m}; \qquad k = 0, 2, 4, \ldots, 2m$$

So,

$$\overline{\epsilon_{conv}} = 0$$

and

$$\overline{\epsilon_{conv}^2} = \frac{2}{2^m} \sum_{k=0}^{m-2} ((m-k) \cdot \epsilon_1)^2 \binom{m}{m-\frac{k}{2}} \tag{7.34}$$

where m is odd and k is even (see Fig. 7.11.). Note that Equations (7.33) and (7.34) are exactly the same except that m is even in Equation (7.33), and it is odd in Equation (7.34). Note also that the error in convolution, for a certain error pattern (where all bits in error have left the first n_1 stages of R_x), is independent of time.

When the bits in error propagate through the first n_1 stages of register R_x, the error in convolution is dependent upon the position of the bits in error and so it is dependent on time. We want to find the signal to noise ratio at those times when the bits in error have passed through register R_{x1}. Assume that we have m bits in error. Therefore, from Appendix A,
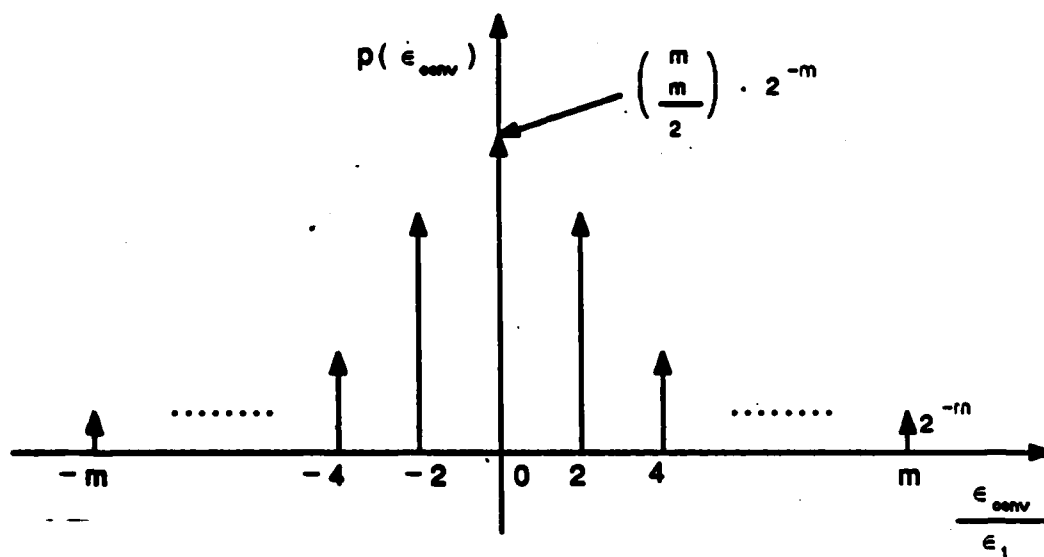
Fig. 7.10. The probability of error in convolution, m even.
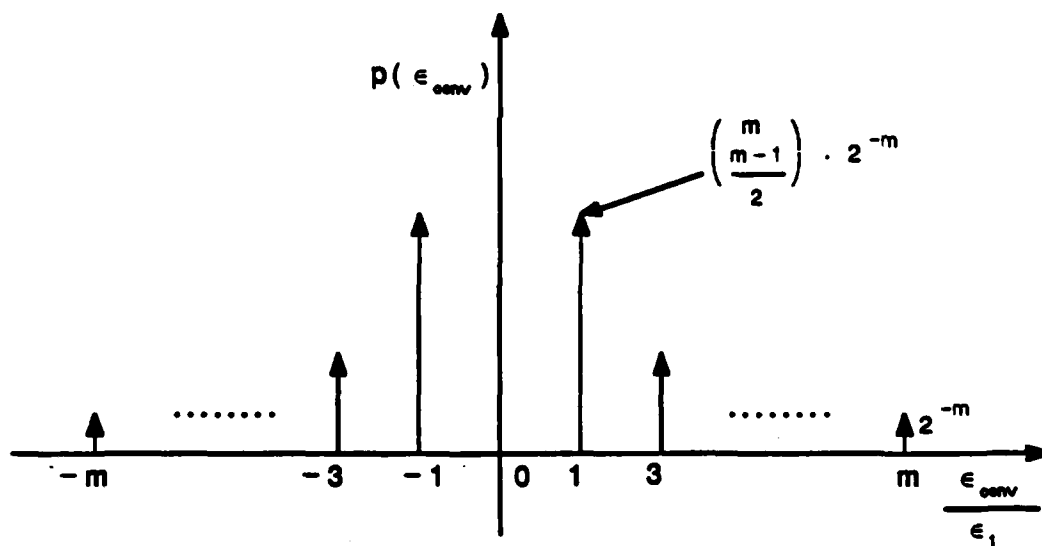
Fig. 7.11. The probability of error in convolution, m odd.

the output $\approx C_1 \cdot \left[\sum_{i=0}^{n_1-1} h_i(n_1 - i) \right] \left[\sum_{i=0}^{j-n_1-1} x_i \right] +$ noise at $t = j \cdot T_s$

where the noise has zero mean and variance given by Equations (7.33) and (7.34). Therefore,

$$SNR \approx \frac{\overline{\left(\frac{c_1}{2} \sum_{i=0}^{j-n_1-1} x_i \right)^2}}{\frac{2c_1^2}{2^m} \sum_{k=0}^{m-2} (m-k)^2 \binom{m}{m-\frac{k}{2}}}$$

$$= \frac{\overline{\left(\sum_{i=0}^{j-n_1-1} x_i \right)^2}}{\frac{8}{2^m} \sum_{k=0}^{m-2} (m-k)^2 \binom{m}{m-\frac{k}{2}}} \tag{7.35}$$

Assume that the probability of a bit in error is p_ϵ. So, the average time period between two bits in error equals $\frac{T_s}{p_\epsilon}$. In practice $p_\epsilon < 10^{-3}$. As a result of this the expected average time between two bits in error is $> 1000 T_s$.

From previous discussions, we know that the effect of a bit that has passed through register R_{x1} is the same as adding dc voltage to the output. As a result of this and because the average time period between two bits in error is greater than $1000 T_s$, an ac coupling in the output of the RDC system will tend to cancel the effect in the output of the bits in error.

E. SIMULATION RESULTS

In this section we present simulation results when the RDC system is used to implement a chirp matched filter and a linear filter (lowpass filter) when additive noise is part of the input.

1. RDC Chirp Matched Filter

The system simulated is shown in block diagram form in Fig. 7.12. The input noise to the lowpass filter is white noise with zero mean and standard deviation sd_1. The bandwidth of the LPF equals that of the chirp signal, and the output band limited noise of the LPF has a standard deviation sd_2, where

$$x(t) = A \sin \left(\frac{2\pi f_2 t^2}{T_d} \right) \tag{7.36}$$

$$f_2 = 5.0 \text{Hz}, T_d = 4.0 \text{ seconds}, F = 8, \text{ and } A = 1.0$$

When the band limited noise has a standard deviation sd_2 equal to 0.75, the average results of six computer runs (where each run had a different seed for the random number generator) are as follows. The maximum value of correlation = 1.641; the output noise has approximately zero mean and a standard deviation 0.2768 where the average chirp signal power numerically is 0.472. See Figures 7.13 and 7.14 for the input and the result of one run. From the given results

$$(SNR)_i = \frac{0.472}{(0.75)^2} = -0.762 \text{ dB}$$

$$(SNR)_o = \left(\frac{1.641}{0.2768} \right)^2 = 15.459 dB$$

Therefore, the signal to noise ratio improvement SNRI = $15.459 - (-0.762)$ = 16.221 dB.

Fig. 7.15 shows the $SNRI$ for different values of sd_2 where theoretically the $SNRI = T_d \cdot B$, and where $T_d \cdot B$ is the pulse duration-bandwidth product of the chirp pulse. (See Horrigan [11].) Hence, for this example, the theoretical $SNRI = 4(5) \Longrightarrow 13\text{dB}$. So, $SNRI$ using the RDC method exceeds the maximum theoretical value by about 3 dB for this case.

Denyer [37] introduced a digital technique called Deltic processing to implement chirp matched filters. We found that the SNRI using the Deltic processing
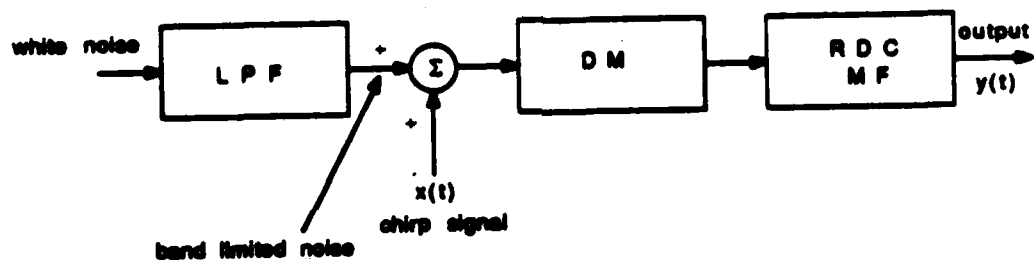
Fig. 7.12. Block diagram of the RDC chirp matched filter that was simulated.
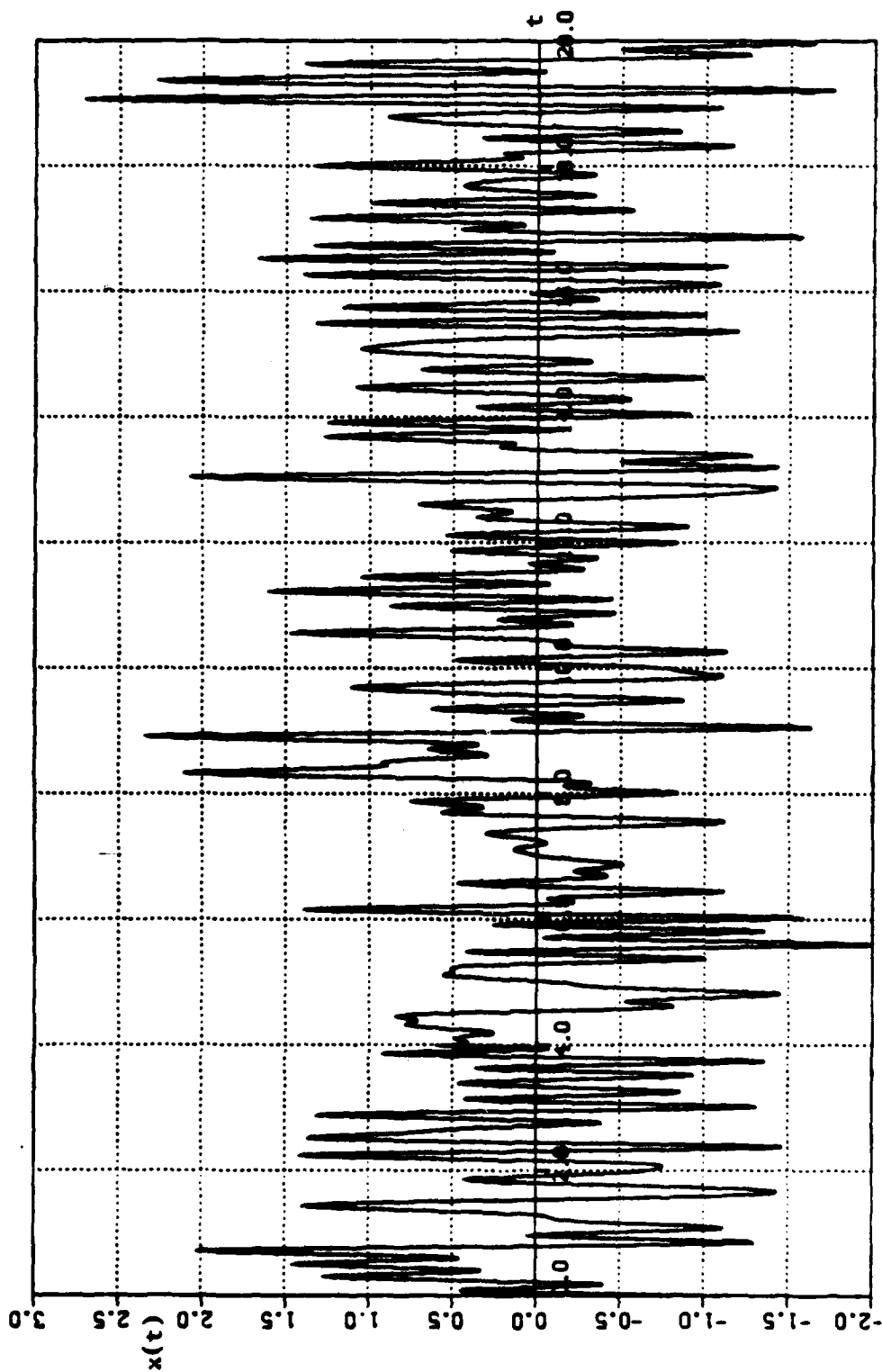
Fig. 7.13. The input of the RDC chirp matched filter.

129
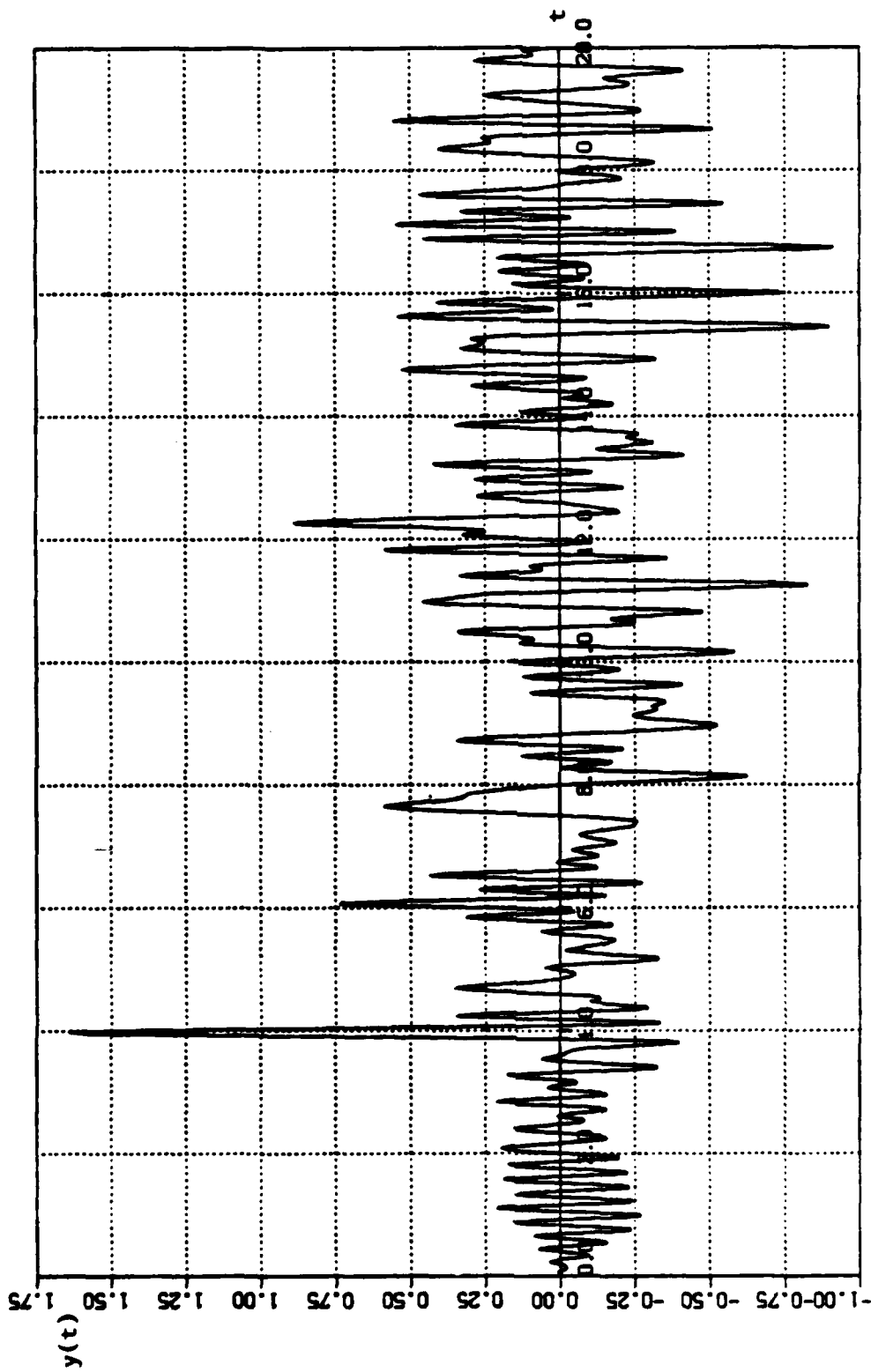
Fig. 7.14. The output of the RDC chirp matched filter.

130
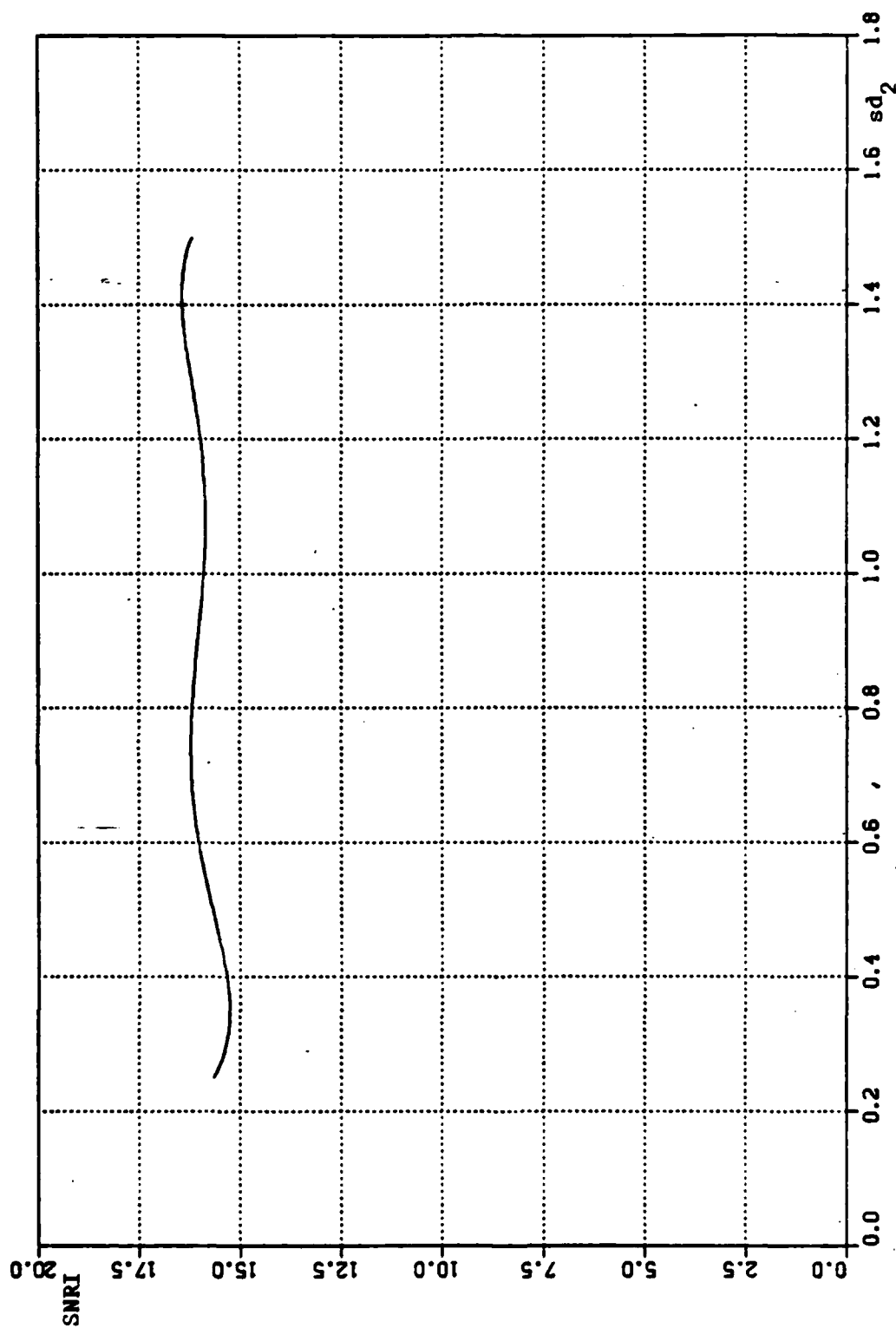
Fig. 7.15. *SNRI* as a function of the standard deviation of the band limited noise when using the RDC system.

technique to implement a chirp MF is less than that obtained by the RDC method by 9 dB.

The noise performance of the RDC system can be explained as follows. In fact, there exists a double match. The first is due to the matching of the filter to the signal. The second is because the delta modulator step size is adjusted to fit the input signal so there is no slope overload. When noise is added to the signal, slope overload occurs and the delta modulator does not always track the signal plus the noise. This lessens the effect of the noise input.

2. Lowpass Filter

In Fig. 7.16, assume that the LPFs have a cut-off frequency f_c equal to 5Hz and that f_1 equals 2.0 Hz where the input is

$$i(t) = x(t) + n(t)$$

where $n(t)$ is zero mean white Gaussian noise with variance $= (0.5)^2$. Assume also that only 5 zeroes at each side of the main lobe of the impulse response of the filters are considered.

Figures 7.17, 7.18, 7.19, and 7.20 show the input $i(t)$, and the outputs $y_1(t), y_2(t)$, and $y_3(t)$. Fig. 7.18 shows that the DM acts like a filter because the step size for the delta modulation process is adjusted to fit the signal. The slope factor is $s = 1.0$. When noise plus signal are applied to the input of the delta modulator, the slope factor s decreases leading to slope overload distortion. This actually reduces the noise effect. In fact, the effective noise power into the RDC system is less than the input power as derived in Equation (7.29).

We can compare the noise performance of the RDC filter with a conventional LPF by defining

132
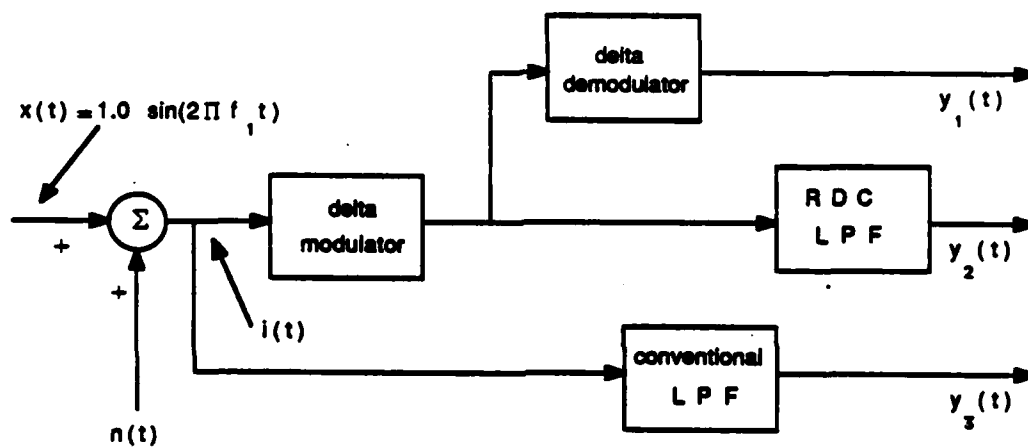
Fig. 7.16. Block diagram of the system used to simulate the outputs of
the delta demodulator, the RDC LPF, and the
conventional LPF.

133
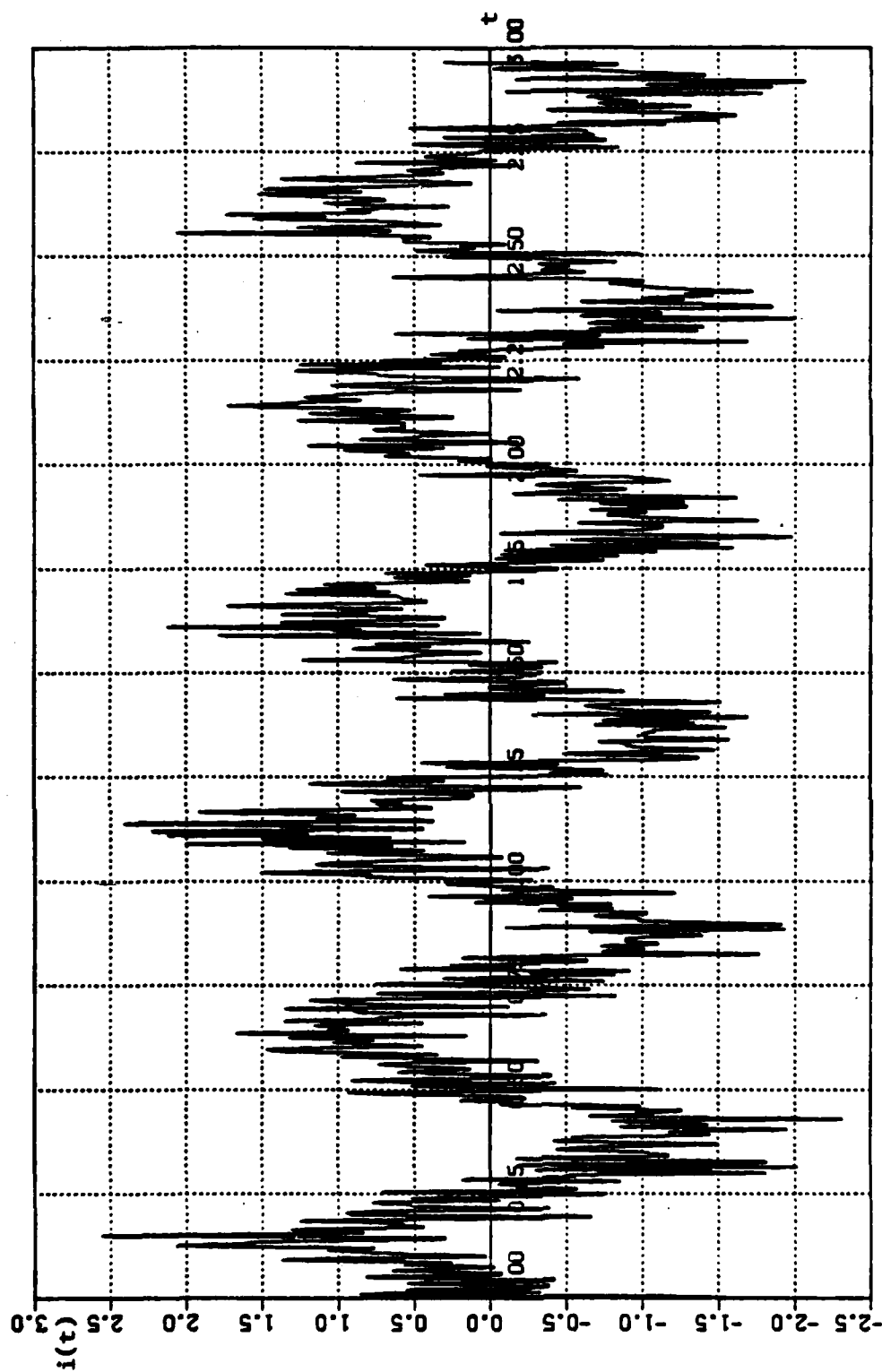
Fig. 7.17. The input to the RDC and conventional lowpass filters.

134
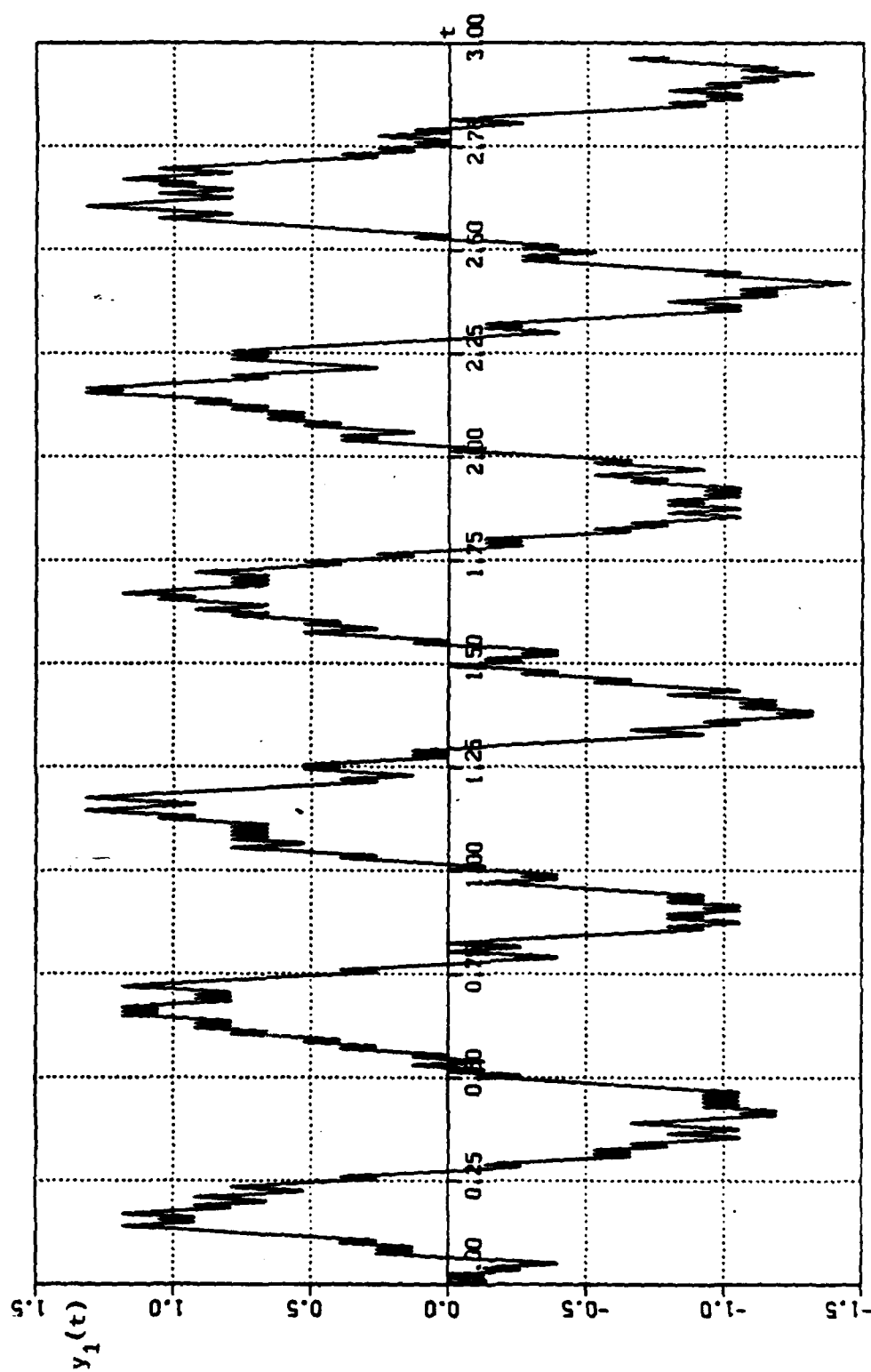
Fig. 7.18. The output of the delta demodulator.

135
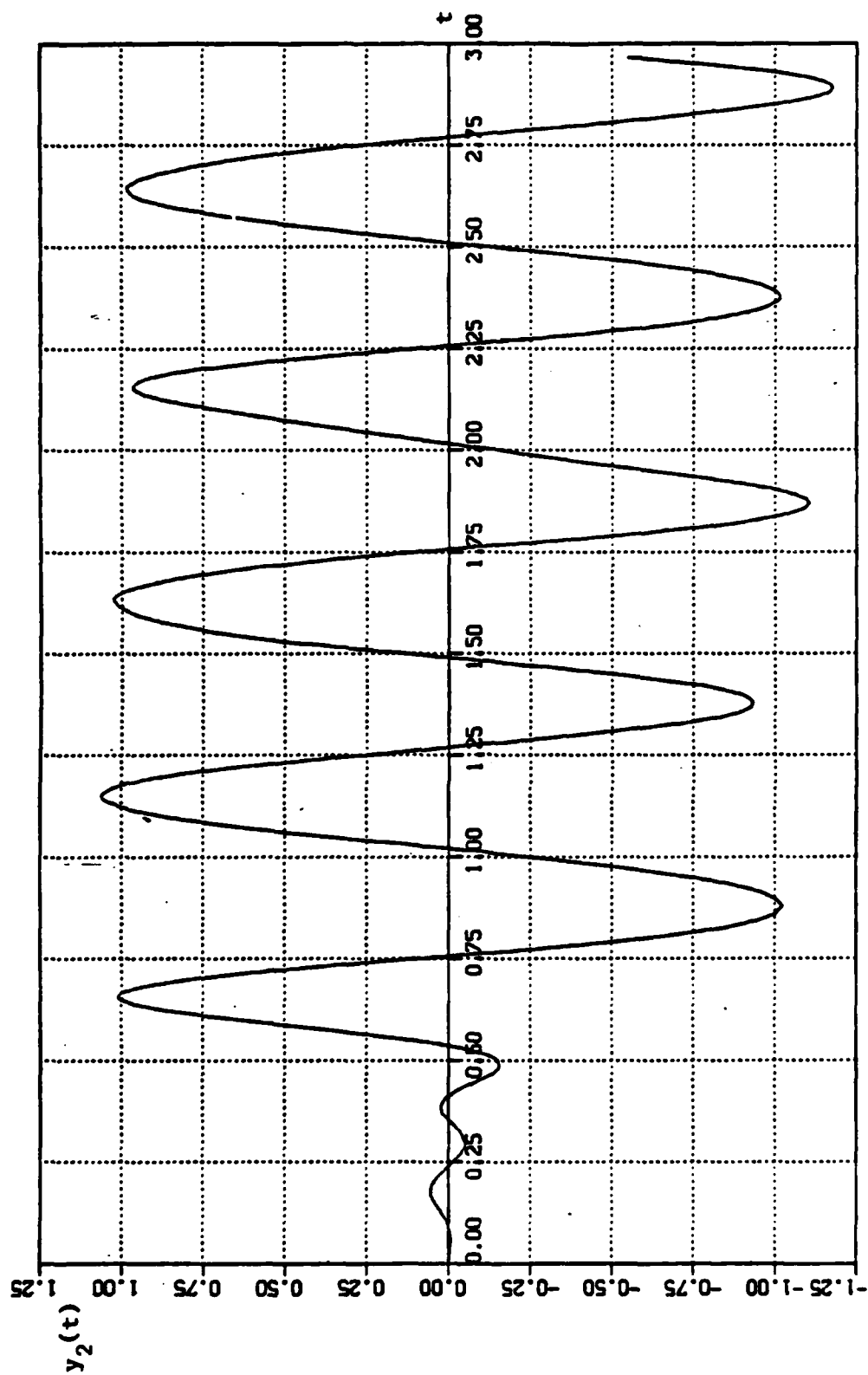
Fig. 7.19. The output of the RDC lowpass filter.

136
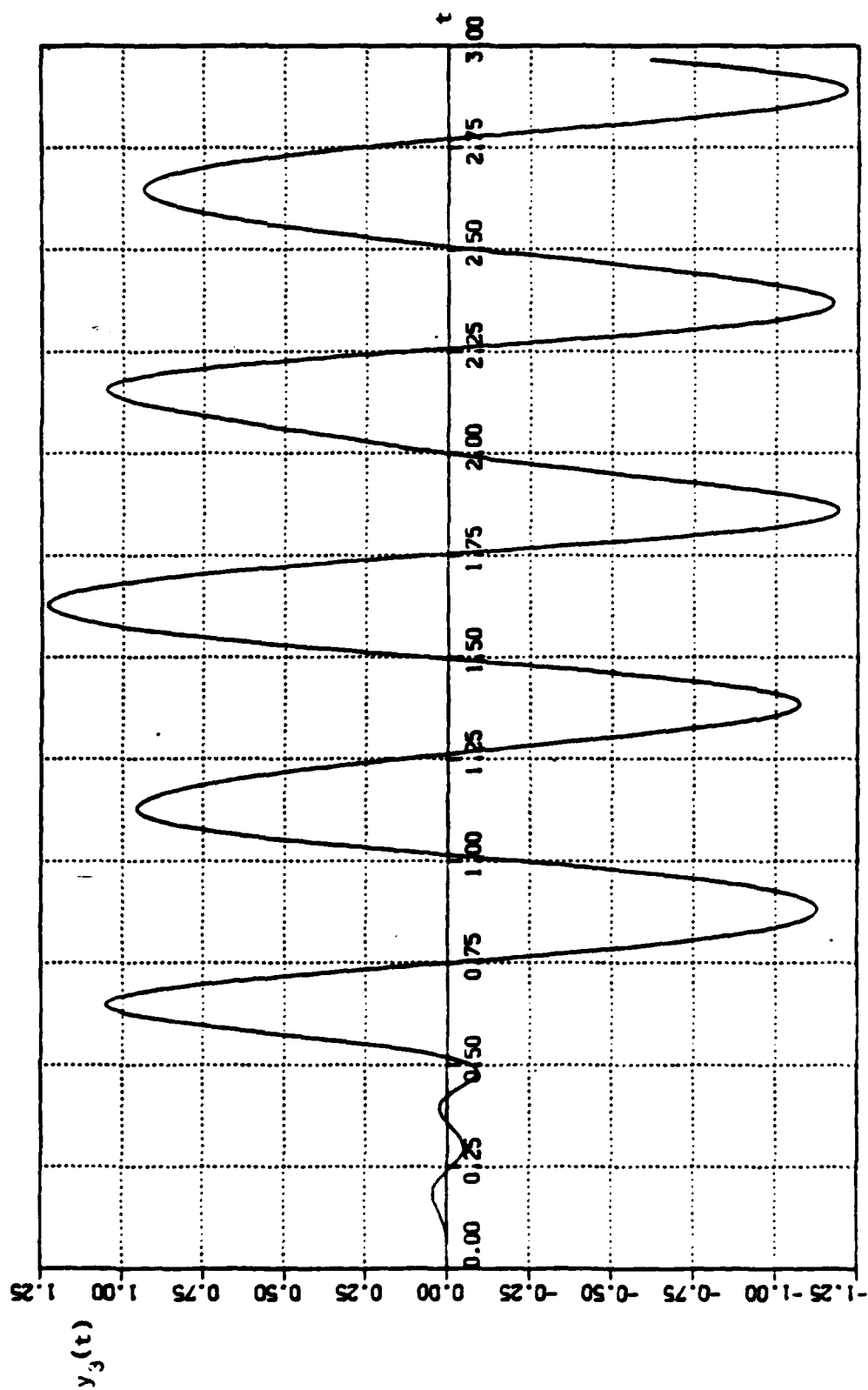
Fig. 7.20. The output of the conventional lowpass filter.

137

$$d_2(t) = y_2(T_{tr} + t) - x(t)$$

and

$$d_3(t) = y_3(t) - x(t)$$

where $T_{tr} \approx 1$ second is the transient time needed to fill the n_1 stages of the register R_{x1} of the RDC system.

The simulation results show that the average power of $d_2(t)$ is equal to 0.01324 where that of $d_3(t)$ is equal to 0.01902 which means that the output of the RDC LPF is closer to the original signal than the output of the conventional LPF. We then conclude that the "effective" noise into the RDC system is less than the input noise of the conventional LPF.

Another way to show that the delta modulator is a "slope" filter is to use the block diagram given in Fig. 7.16 to find r_1, r_2, and r_3 versus σ_i^2 where

$$r_1 = \frac{\sigma_1^2}{\sigma_i^2}$$

$$r_2 = \frac{\sigma_2^2}{\sigma_i^2}$$

$$r_3 = \frac{\sigma_3^2}{\sigma_i^2}$$

and where $\sigma_1^2, \sigma_2^2, \sigma_3^2$, and σ_i^2 are the variances of $y_1(t), y_2(t), y_3(t)$ and $i(t)$ respectively. We consider the case where the input $i(t)$ is white Gaussian noise only and where the step size of the delta modulator is chosen to fit a sinusoid of amplitude 1 volt and frequency equal to the bandwidth of the LPF which is assumed to be 5 Hz.

Simulation results are summarized in Fig. 7.21 as plots of r_1, r_2, and r_3 versus σ_i^2. Note

138
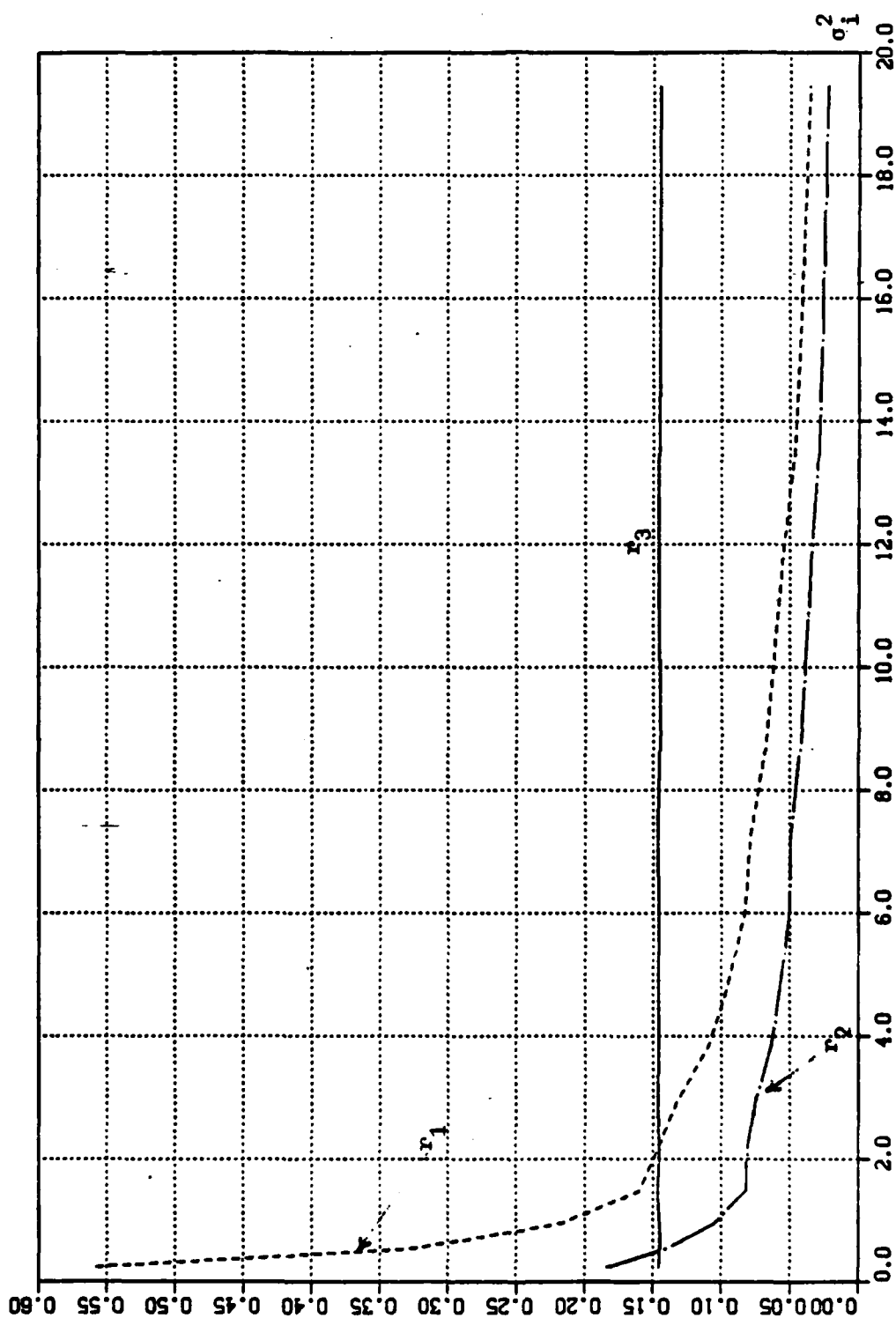
Fig. 7.21. The values of $r_1, r_2,$ and r_3 as a function of σ_i^2.

139

- For the RDC LPF, r_1 and r_2 decrease as the variance of the input increases. This is because an increase in the variance of the input creates additional slope overload conditions. The delta modulator then acts as a "slope" filter. The slope of the output of the delta demodulator increases but at a smaller rate. As a result of this the values of r_1 and r_2 decrease as σ_i^2 increases.

- When σ_i^2 is small compared to the step size, granular noise occurs and so r_1 will be greater than r_3.

The simulation results verify that the noise performance of the RDC system is better than that expected from the theory of linear filters. The reason for this is that the DM portion of the RDC system filters on the basis of slope as well as frequency of the applied voltage.

F. USE OF VARIABLE STEP SIZE WITH THE RDC METHOD TO OVERCOME GRANULAR NOISE OF THE IMPULSE RESPONSE REPRESENTATION

Assume that we want to represent the impulse response $h(t)$ for a LPF or a band suppression filter using DM where the step size Δh is constant. For a given sampling rate, if Δh is appropriate for the main lobe (to prevent slope overload) then the DM values for all side lobes whose amplitudes are less than the step size are successive values of $+1$ and -1. This means that the side lobes are represented as being constant. On the other hand, if we choose the step size Δh appropriate for the side lobes, the DM cannot track the main lobe without increasing the sampling frequency. This requires an increase in the number of stages of the two shift registers R_h and R_x. One solution is to use different step sizes for different regions of $h(t)$. We call this the variable step size RDC method.

1. Implementation of the Variable Step Size RDC Method

To implement the variable step size RDC method, we first define $h(t)$ in the range $\pm \frac{n_1}{2} T_s$ seconds. To make the impulse response causal, shift the time axis by $\frac{n_1}{2} T_s$ seconds. An example is shown in Fig. 7.22.

Now divide the causal impulse response into M different time intervals so that in each interval a single step size is appropriate. In Fig. 7.22 $M = 5$ and, because of symmetry, the step size for region A equals that of E and the step size for region B equals that of D. Assume that the M intervals have $n_1, n_2, \ldots, n_M$ samples and step sizes $\Delta h_1, \Delta h_2, \ldots, \Delta h_M$, respectively.

Using the procedure given in Appendices A and B, it is easy to show that

$$
\begin{aligned}
\Delta 2conv\,(j \cdot T_s) = \Bigg\{ &\sum_{i=0}^{n_1-1} [<R_h(i)> \cdot <R_x(i)>] \cdot \Delta h_1 \\
+ &\sum_{i=n_1}^{n_1+n_2-1} [<R_h(i)> \cdot <R_x(i)>] \cdot \Delta h_2 \\
+ &\sum_{i=n_1+n_2}^{n_1+n_2+n_3-1} [<R_h(i)> \cdot <R_x(i)>] \cdot \Delta h_3 \\
+ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
+ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
+ &\sum_{i=N-n_M}^{N-1} [<R_h(i)> \cdot <R_x(i)>] \cdot \Delta h_M \\
- &<R_x(N)> \cdot k_5 \Bigg\} \Delta x \cdot T_s
\end{aligned}
\tag{7.37}
$$

where

$$
N = \sum_{i=1}^{M} n_i
$$

and

$$
k_5 = \Delta h_1 \cdot \sum_{i=0}^{n_1-1} h_i + \Delta h_2 \cdot \sum_{i=n_1}^{n_1+n_2-1} h_i + \ldots\ldots + \Delta h_M \cdot \sum_{i=N-n_M}^{N-1} h_i
$$

$=$ constant for a given impulse response and given step sizes.

141
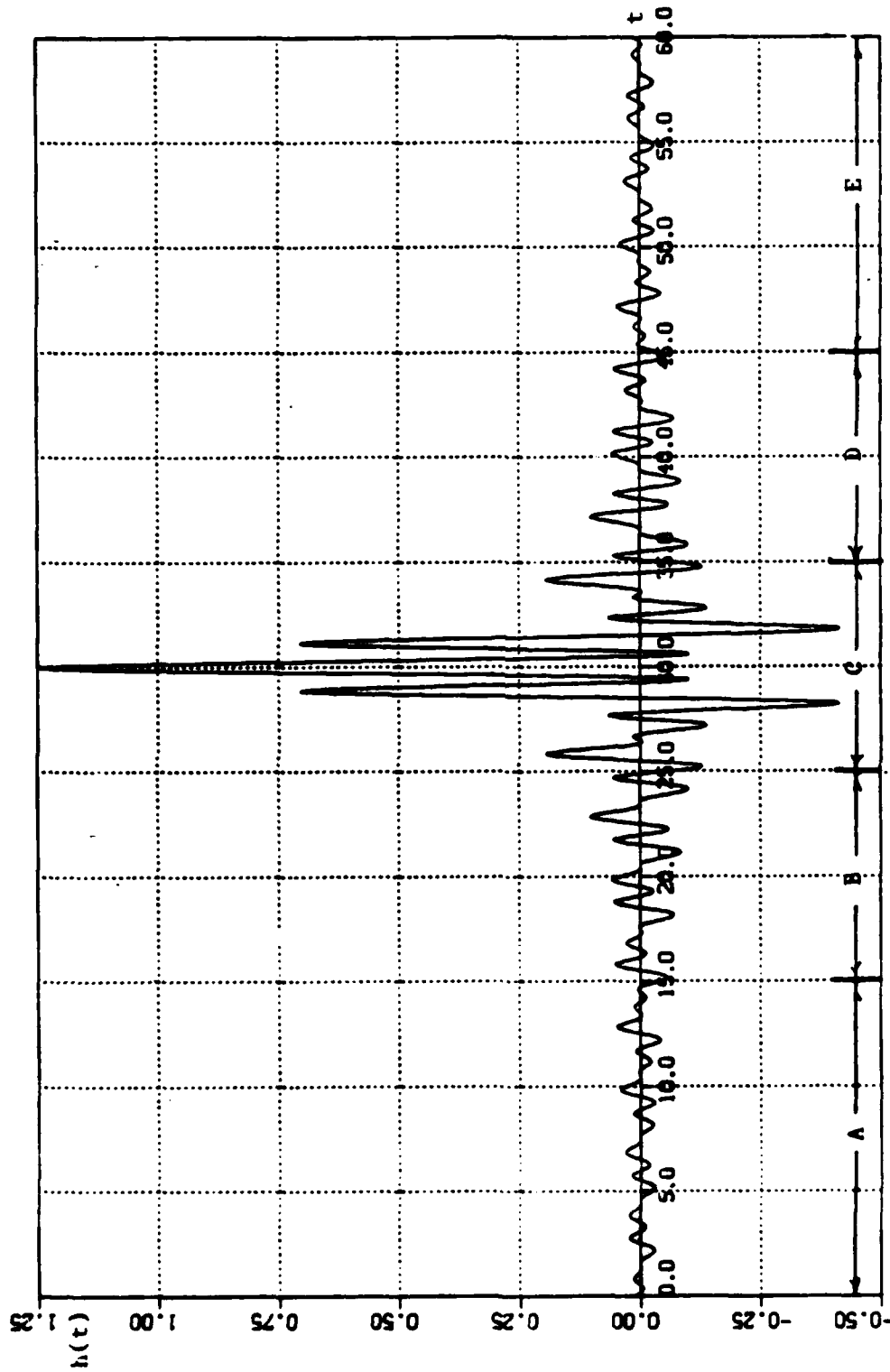
Fig. 7.22. The causal impulse response of a band suppression filter divided into 5 regions, each of which has an appropriate step size.

Equation (7.37) can be implemented as shown in Fig. 7.23 where the constant $\Delta x \cdot T_s$ can be off the IC chip.

2. Applications

This section presents simulation results obtained using the variable step size RDC method to implement a lowpass filter and a band suppression (notch) filter.

a. Variable Step Size RDC Lowpass Filter

Assume that the cutoff frequency of the LPF is 1.0 Hz. Figures 7.24, 7.25 and 7.26 show the impulse response and the staircase representations ($h_v(t)$ and $h_{ct}(t)$) when we use variable and constant step sizes and where we take into consideration 40 zeroes on each side of the main lobe. The number of intervals is $M = 21$. Fig. 7.27 shows the resulting transfer function when we use variable step size. Also shown is the transfer function when a constant step size is used. Using the variable step size RDC method, the ripple in the transfer function is reduced and the filter skirts are steeper.

b. Variable Step Size RDC Band Suppression (Notch) Filter

Fig. 7.28 shows the transfer function of a band suppression filter where

$$f_b = 1.0\text{Hz}, f_a = \frac{f_b}{2} = 0.5, \text{ and } \Delta f = 0.375.$$

and where

$$H_n(f) = H_l(f) - H_b(f)$$

Figures 7.29, 7.30 and 7.31 show the impulse response and the corresponding staircase representation when we use the variable step size and constant step size RDC method. The number of intervals for the variable step size is $M = 17$. The transfer functions obtained are shown in Fig. 7.32.

From Figures 7.27 and 7.32, we can say that the variable step size RDC method has a better transfer function than that of the constant step size RDC method.

143
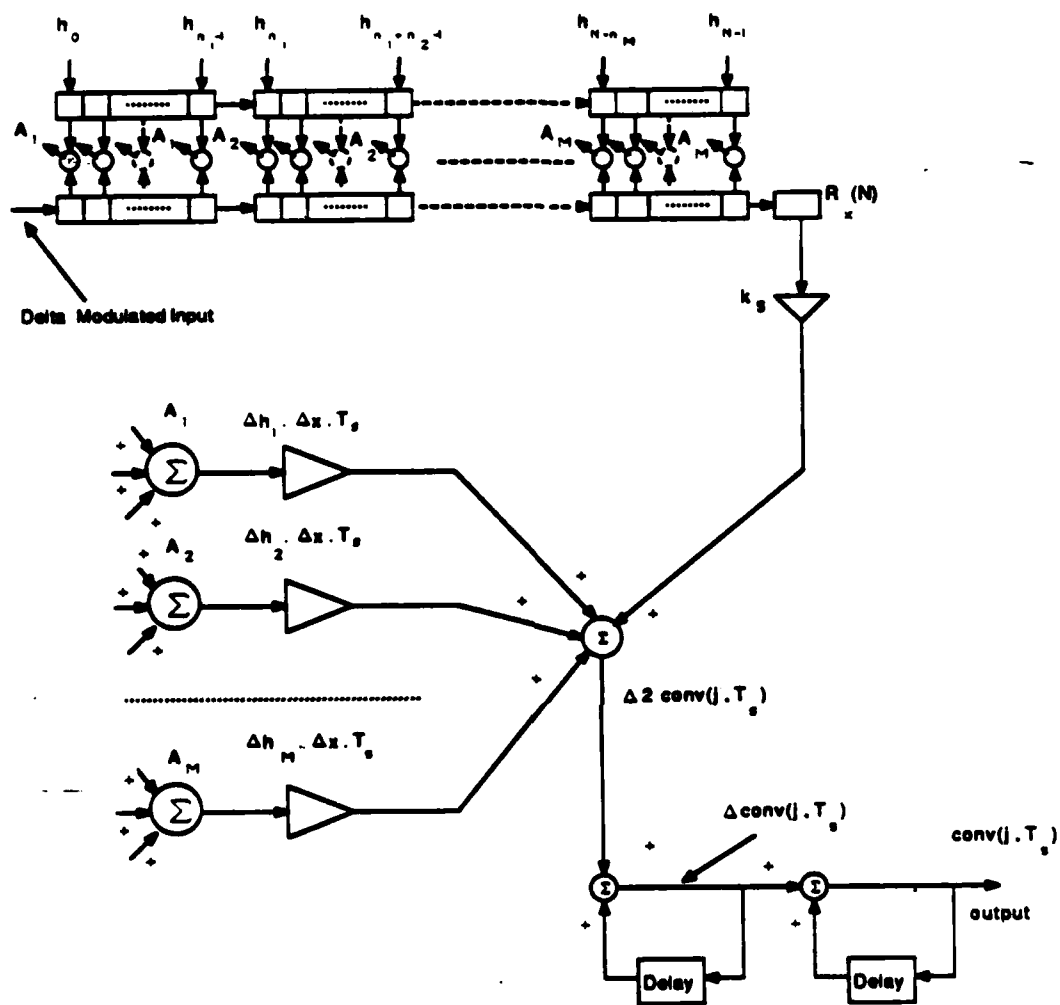
Fig. 7.23. Diagram of the variable step size RDC system.

144
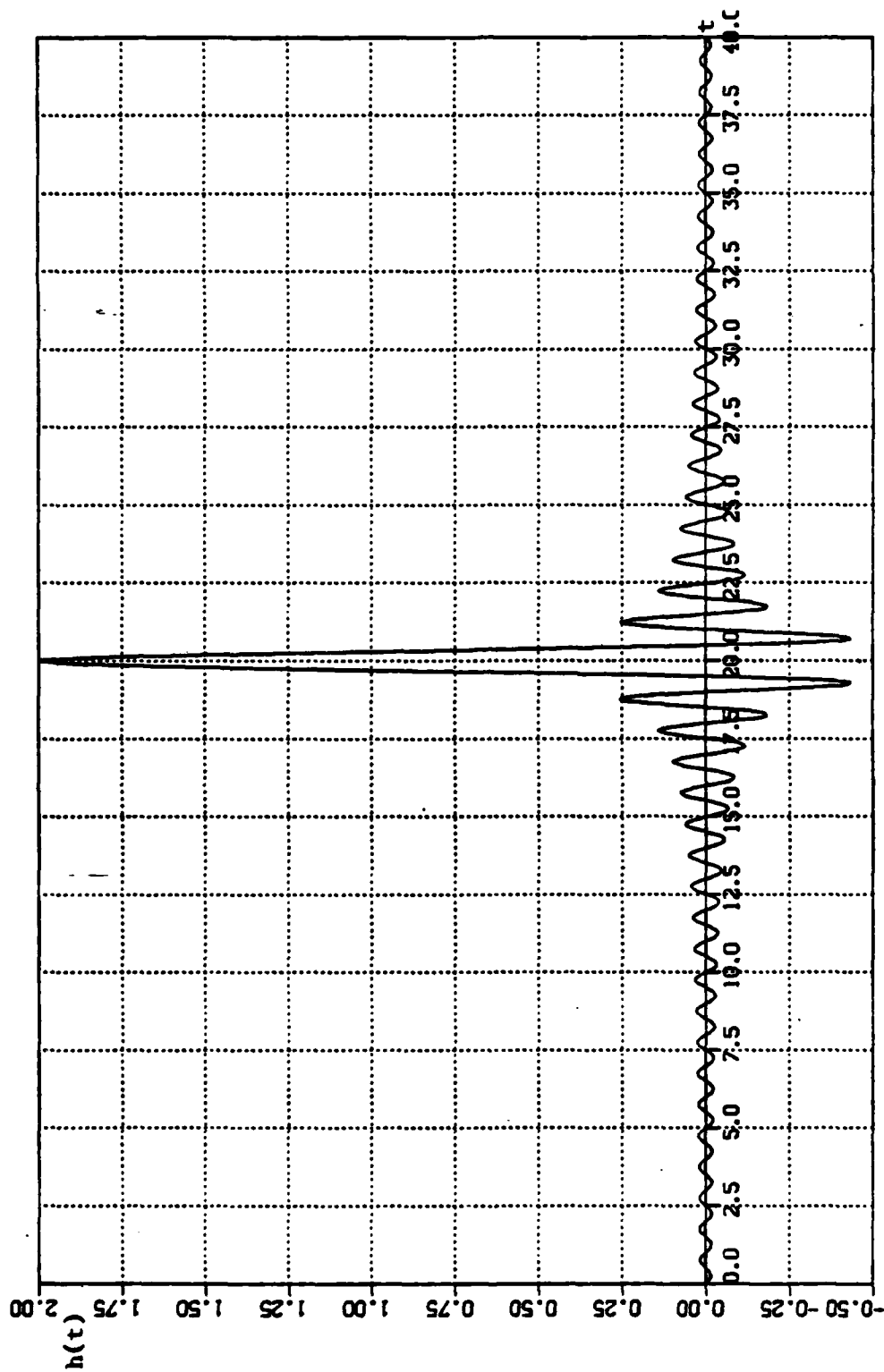
Fig. 7.24. The LPF causal impulse response.
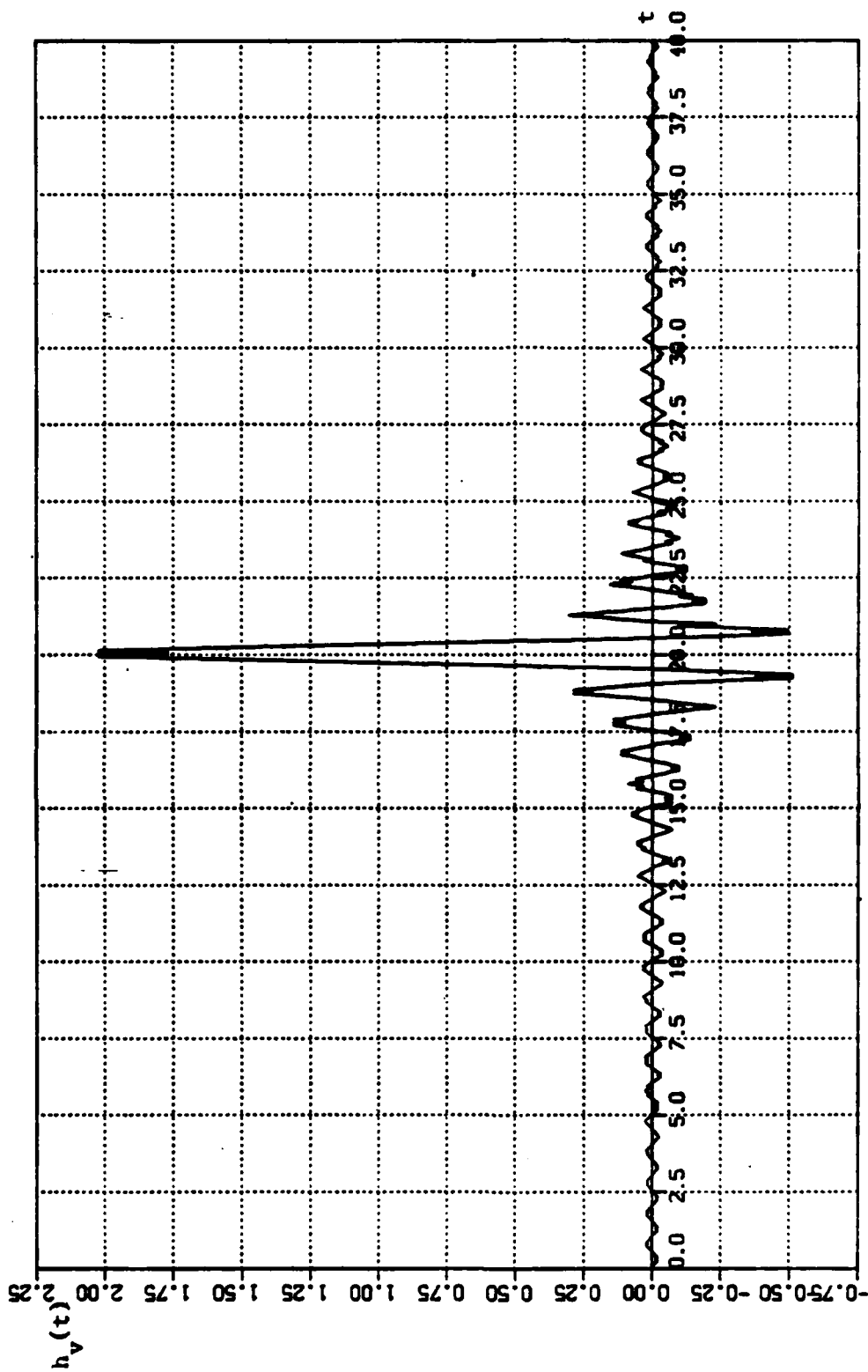
Fig. 7.25. The staircase representation of the causal impulse response of the LPF when the variable step size is employed.
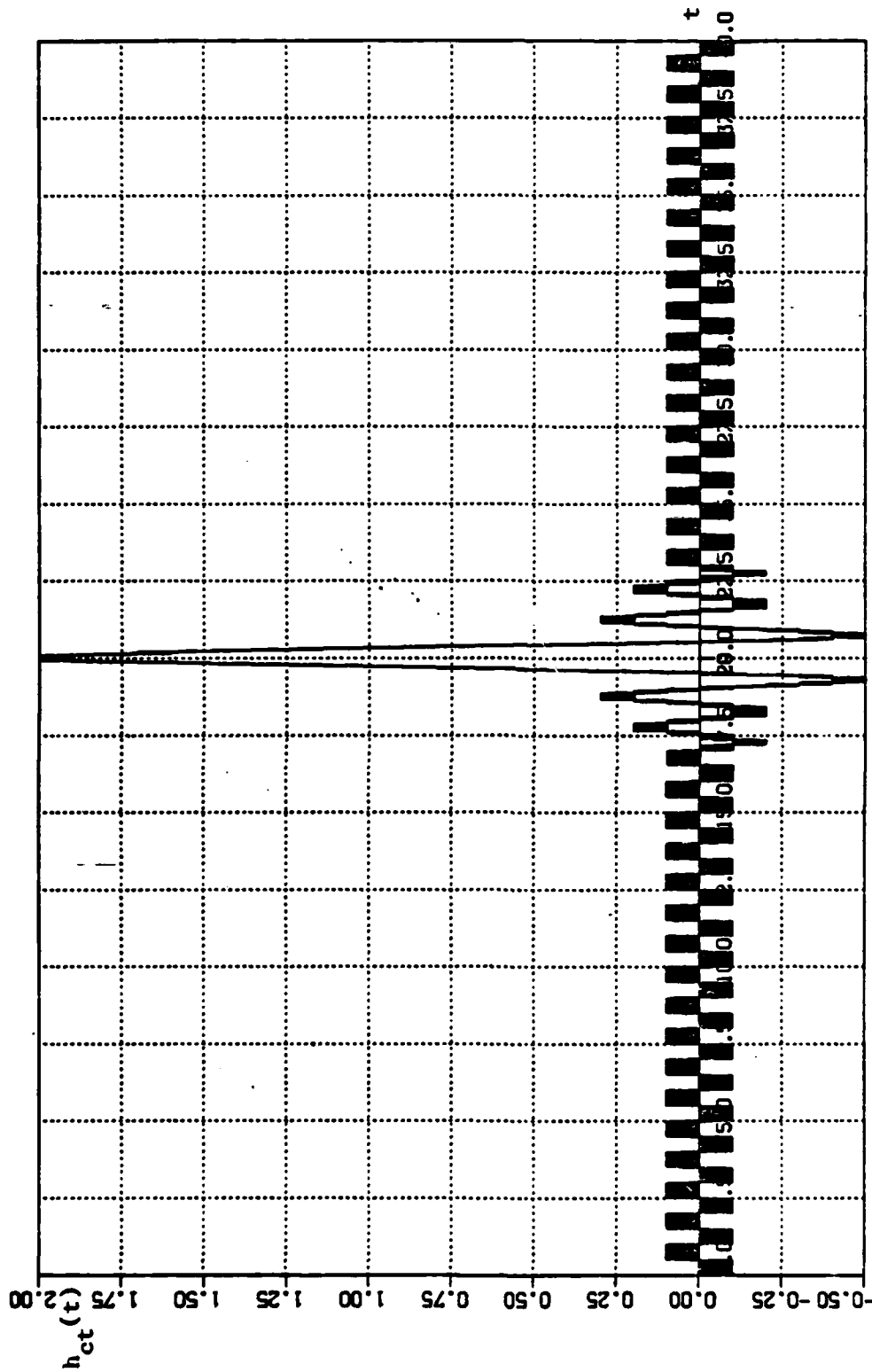
Fig. 7.26 The staircase representation of the causal impulse response of the LPF when the constant step size is employed.

147
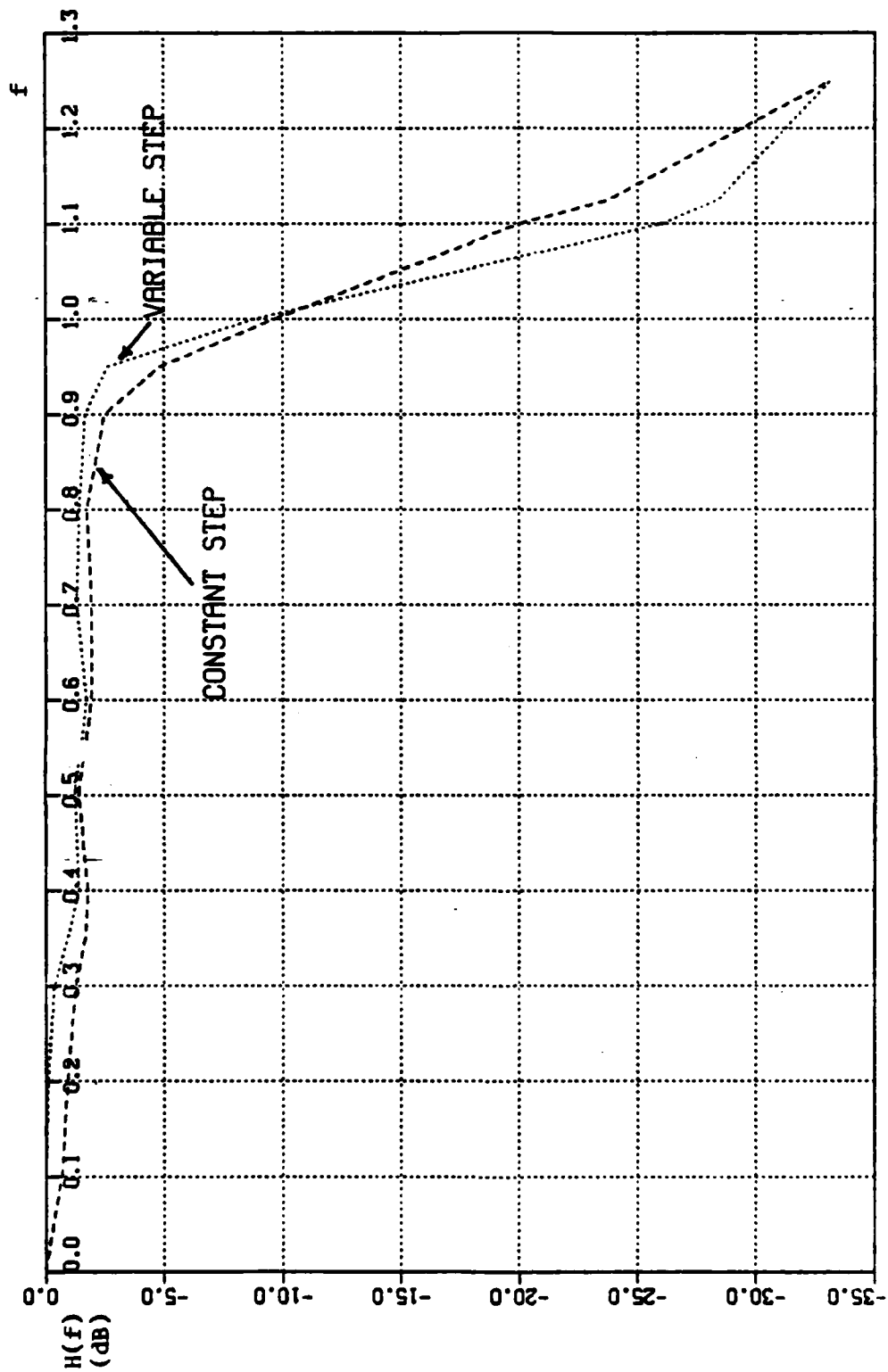
Fig. 7.27. Transfer functions of constant and variable step size RDC LPF's.

148
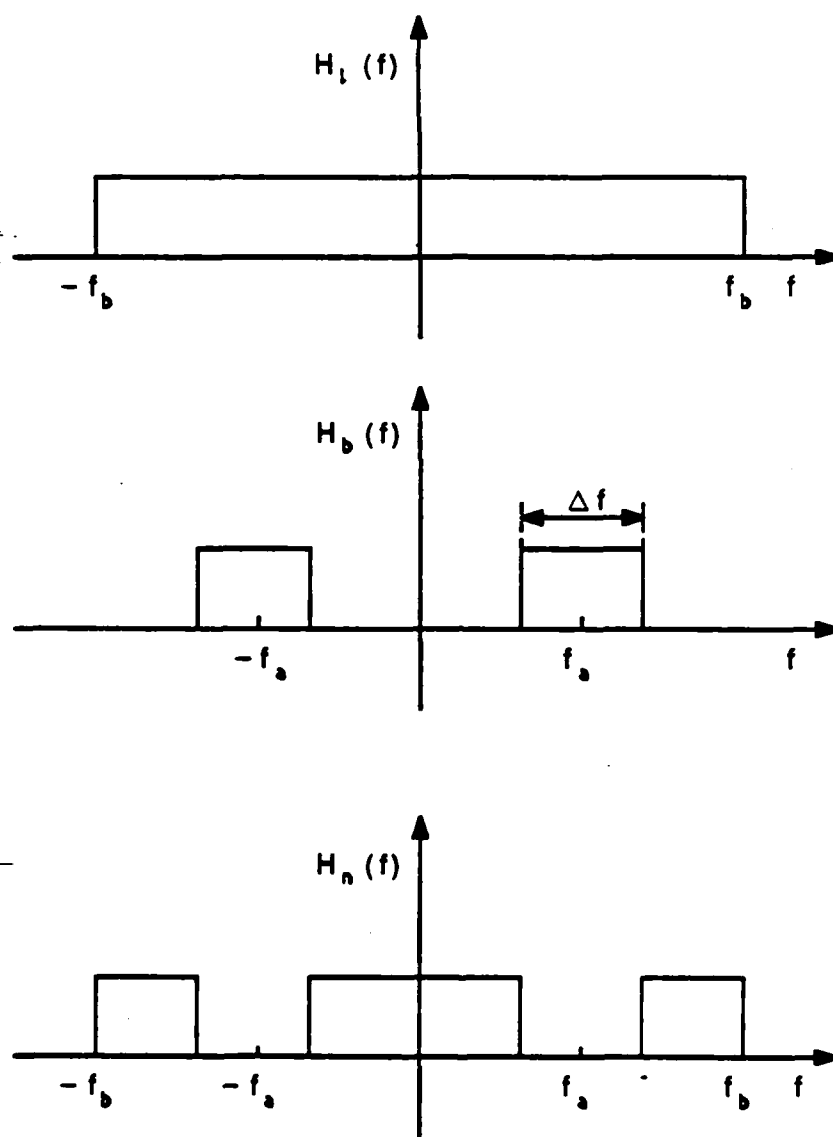
Fig. 7.28 A band suppression filter transfer function $H_n(f)$ where
$$H_n(f) = H_l(f) - H_b(f)$$

149
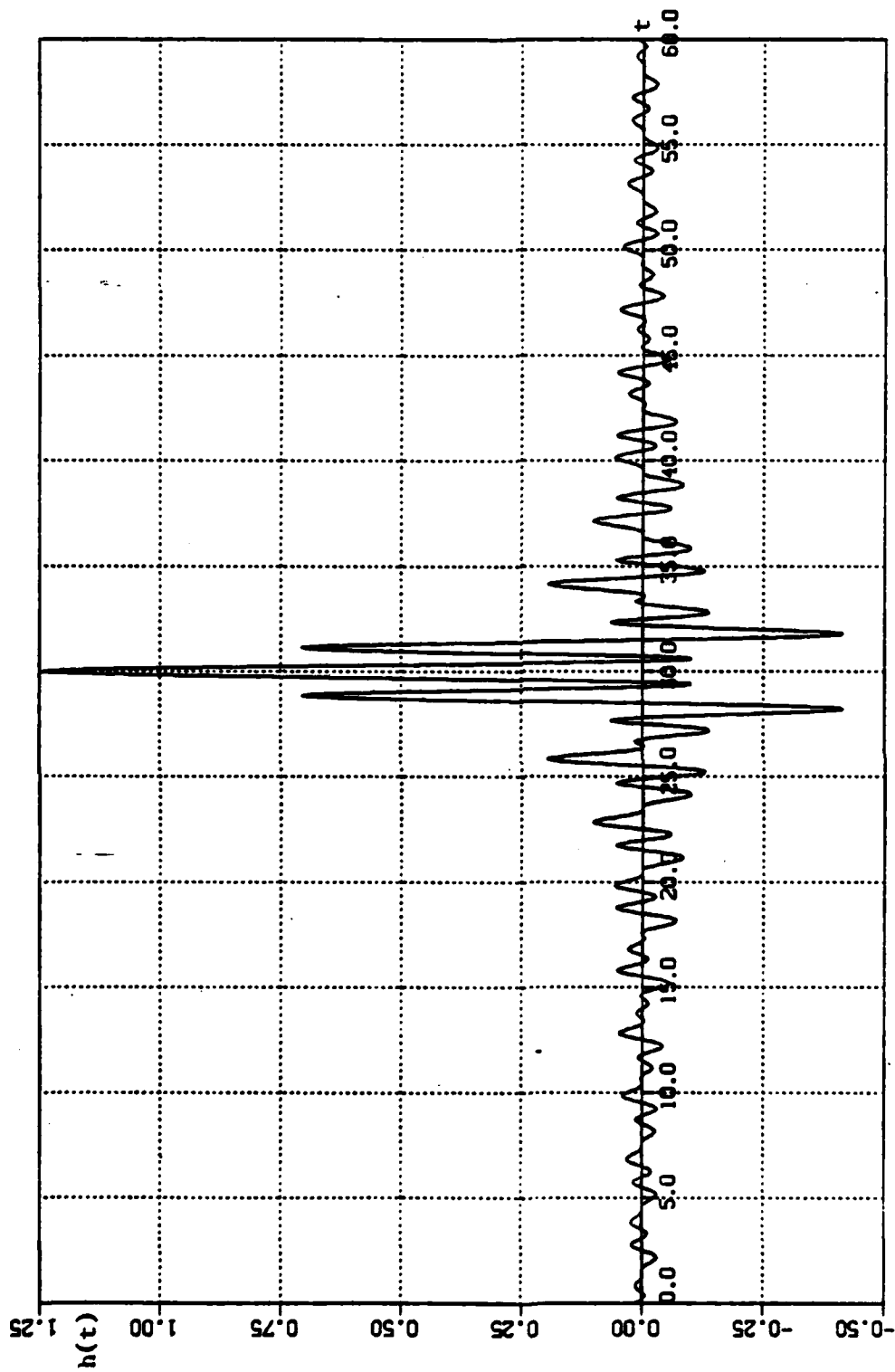
Fig. 7.29. The impulse response of the band suppression filter.

150
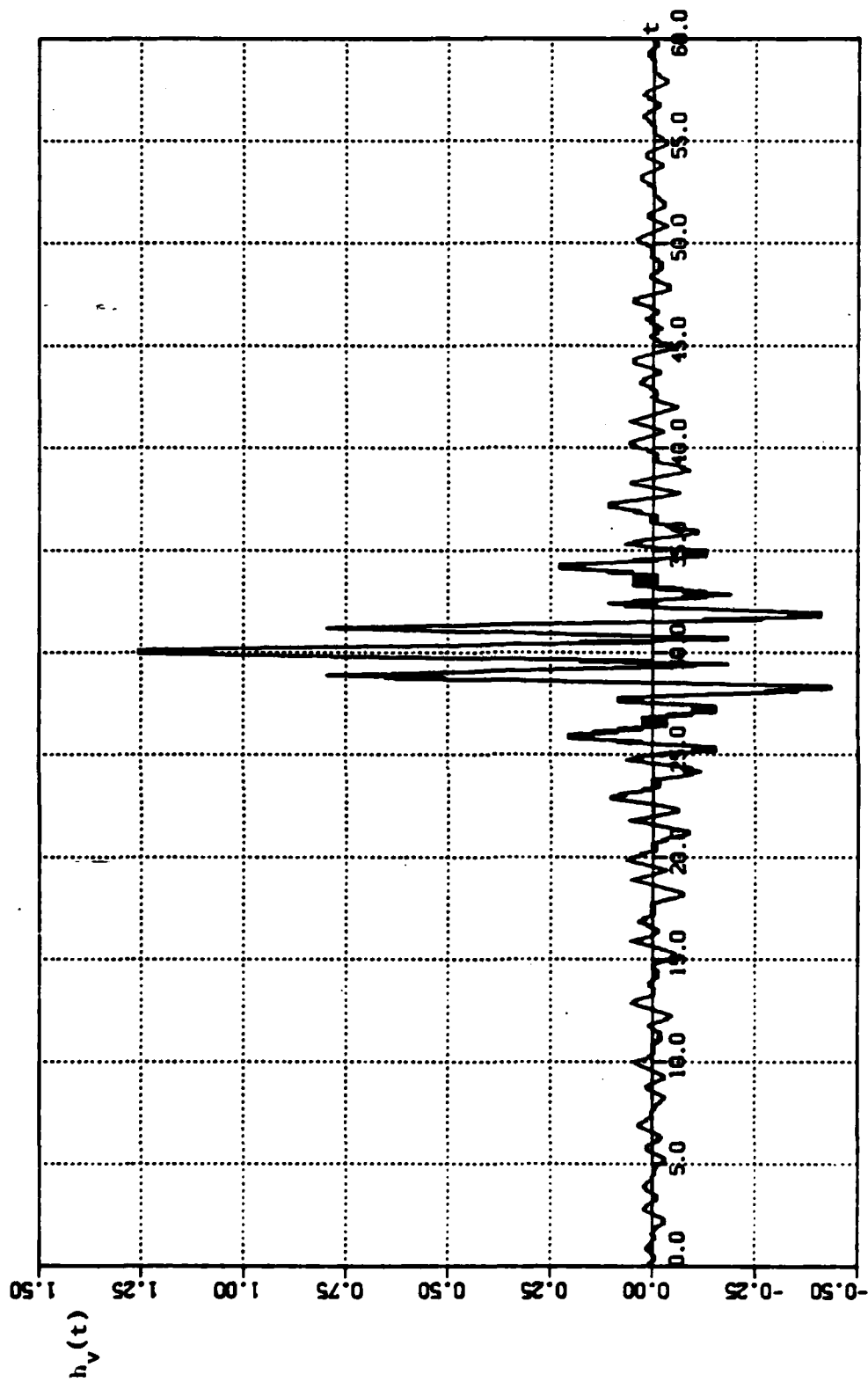
Fig. 7.30. The staircase representation of the impulse response of the band suppression filter when variable step size is employed.[1]

151
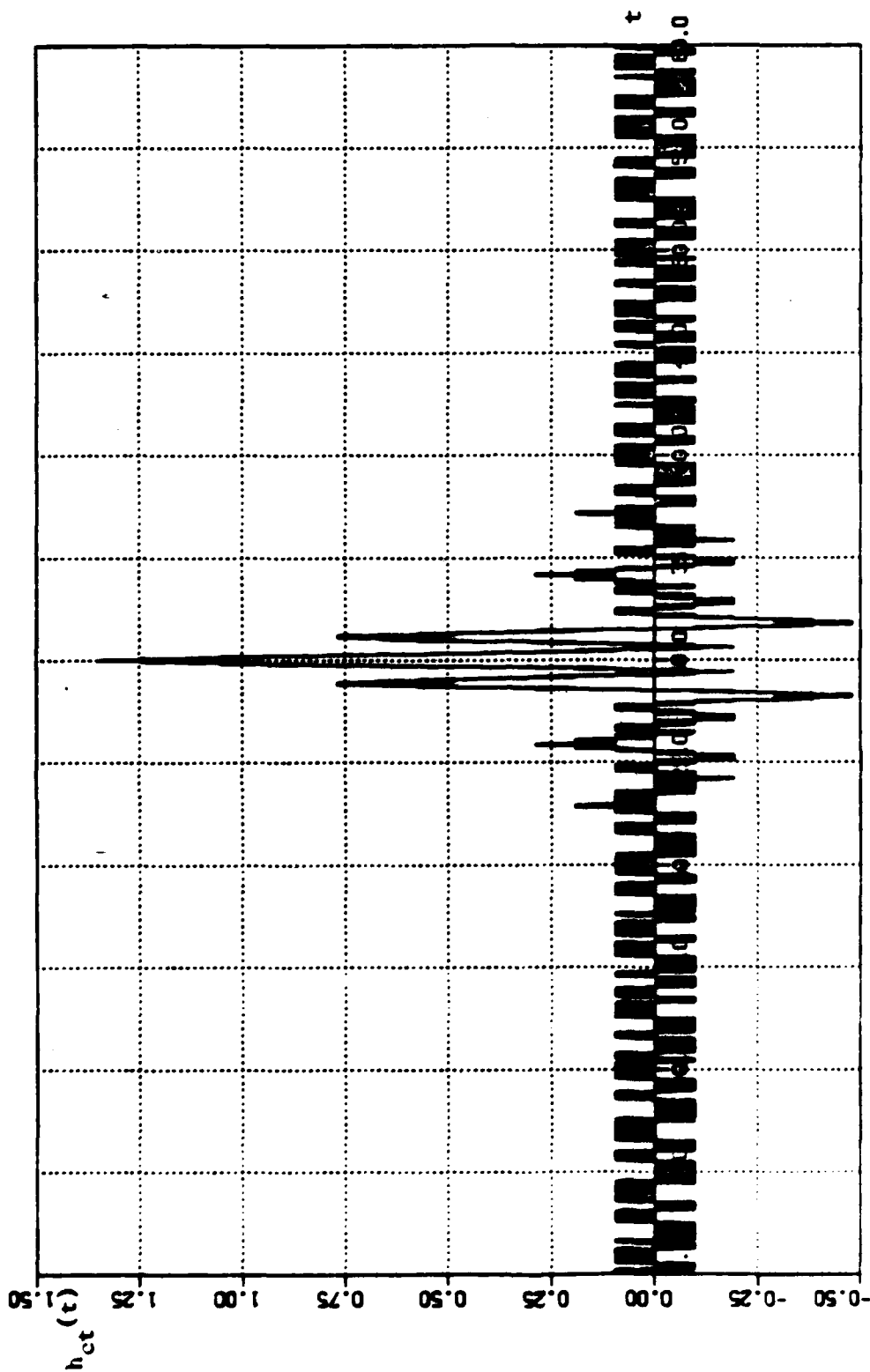
Fig. 7.31. The staircase representation of the impulse response of the band suppression filter when constant step size is employed.

152
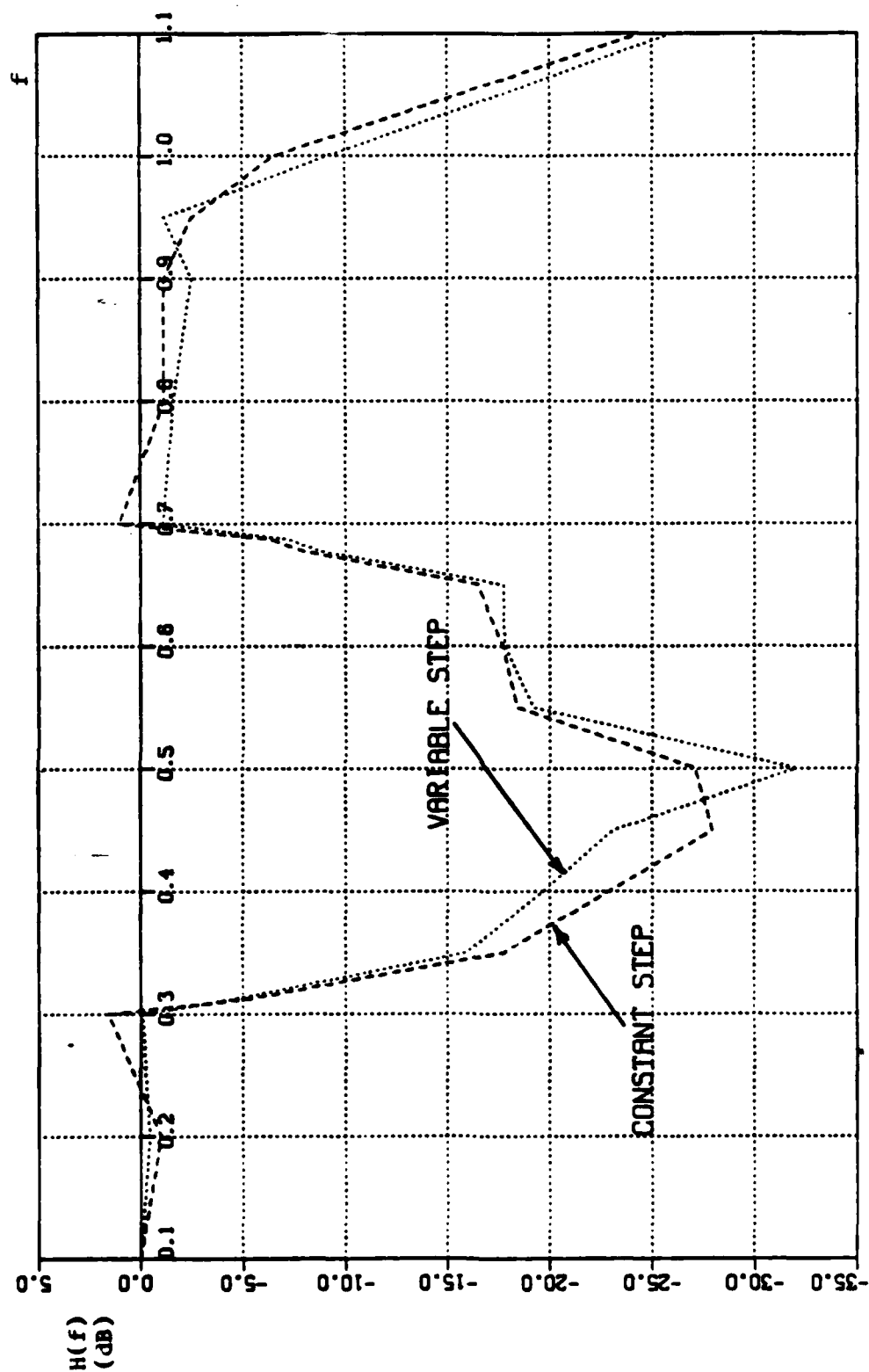
Fig. 7.32. Transfer functions of constant and variable step size RDC notch filters.

153

VIII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The RDC system is a new realization of digital filters. The system can be used as a convolver or as a correlator. The RDC system has a simple layout for IC realization. When compared with other digital filter realizations, the RDC system appears to require less hardware and fewer connections. The system requires no synchronization, has no multipliers and is easily programmable. The RDC system provides real time operation (no off-line processing). There is no limitation on the time duration of the signal to be represented in matched filter applications. Simulation results indicate the RDC system achieves noise performance which surpasses that predicted by the theory of linear filters.

B. RECOMMENDATIONS

It is recommended that a RDC system be fabricated and tested to confirm simulation results and to provide additional insight into this method of achieving convolution.

It is also recommended that additional study be done and the results be extended to include adaptive delta modulation techniques.

Also, studies of other noise reduction methods including additional work on slope filtering are recommended.

APPENDIX A

IMPLEMENTATION OF A TRANSVERSAL FILTER
USING DELTA CONVOLUTION

Assume that we want to convolve the analog pulse signals $h(t)$ and $x(t)$ where $\hat{h}(t)$ and $\hat{x}(t)$ are the delta modulation versions of $h(t)$ and $x(t)$. Assume that $\hat{h}(t)$ and $\hat{x}(t)$ have number of bits n_1 and n_2 respectively where

$$n_1 < n_2$$

Assume also that we have two registers R_h and R_x where R_h holds the delta modulation bits of $h(t)$, and R_x holds the delta modulation bits of $x(t)$ which have to be convolved with the contents of $R_h \cdot So, R_h$ has n_1 memory stages and R_x has n_2 stages.

By expanding Equation (3.18) we can write the result of convolution at time $j \cdot T_s$ as

$$
\begin{aligned}
y(t) \approx y(j \cdot T_s) = C_1 \cdot \Bigg\{ &h_0 \left[\sum_{i=0}^{j-1} x_i + \sum_{i=0}^{j-2} x_i + \sum_{i=0}^{j-3} x_i + \ldots + \sum_{i=0}^{j-n_1} x_i \right] \\
&+ h_1 \left[\sum_{i=0}^{j-2} x_i + \sum_{i=0}^{j-3} x_1 + \ldots + \sum_{i=0}^{j-n_1} x_i \right] \\
&+ h_2 \left[\sum_{i=0}^{j-3} x_i + \ldots + \sum_{i=0}^{j-n_1} x_i \right] \\
&+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
&+ h_{n_1-1} \left[\sum_{i=0}^{j-n_1} x_i \right] \Bigg\}
\end{aligned}
$$

155

which can be rewritten as

$$y(t) \approx y(j \cdot T_s) = C_1 \cdot \left\{ h_0 \left[x_{j-1} + 2x_{j-2} + 3x_{j-3} + \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \right.\right.$$

$$\left. + n_1 \cdot x_{j-n_1} + n_1 \cdot \sum_{i=0}^{j-n_1-1} x_i \right]$$

$$+ h_1 \left[\quad x_{j-2} + 2x_{j-3} + \dots\dots\dots\dots\dots\dots\dots\dots\dots \right.$$

$$\left. + (n_1 - 1) \cdot x_{j-n_1} + (n_1 - 1) \cdot \sum_{i=0}^{j-n_1-1} x_i \right]$$

$$+ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$

$$+ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots$$

$$\left. + h_{n_1-1} \left[\sum_{i=0}^{j-n_1-1} x_i \right] \right\}$$

$$(A.1)$$

The products needed to implement Equation (A.1) can be represented as shown in Fig. A.1. A schematic diagram of a system which provides $y(j \cdot T_s)$ according to Equation (A.1) is shown in Fig. A.2 where f_s is the sampling frequency.

In Fig. A.2 we assume that R_A register stages can have any of three states $-1, 0$, or $+1$ and that it is bidirectional. Assume also that we have a logic which can do the following:

a) During the time periods between even and odd sampling pulses the contents of register R_A move to the right.

b) During the time periods between odd and even sampling pulses the contents of register R_A move to the left.

Register R_A has $2n_1$ stages. The first n_1 stages have to be initialized with the delta modulation version of $h(t)$ which can be either $+1$ or -1. The last n_1 stages of register R_A have to be initialized with zeroes.

156
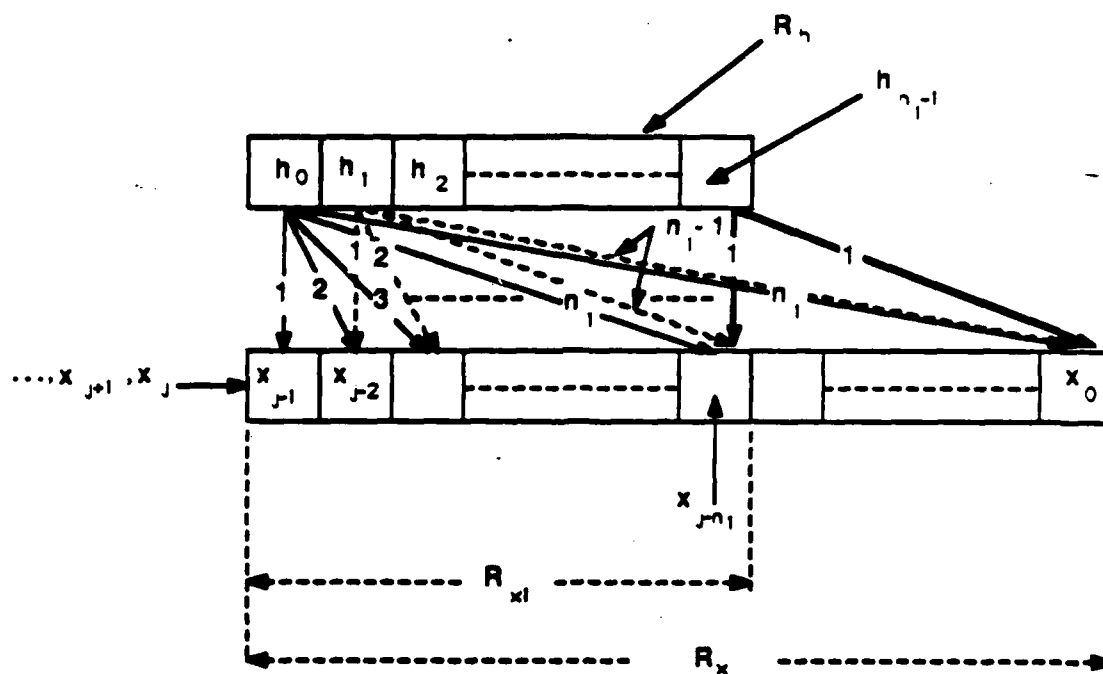
Fig. A.1. Diagram showing the products needed for delta convolution.
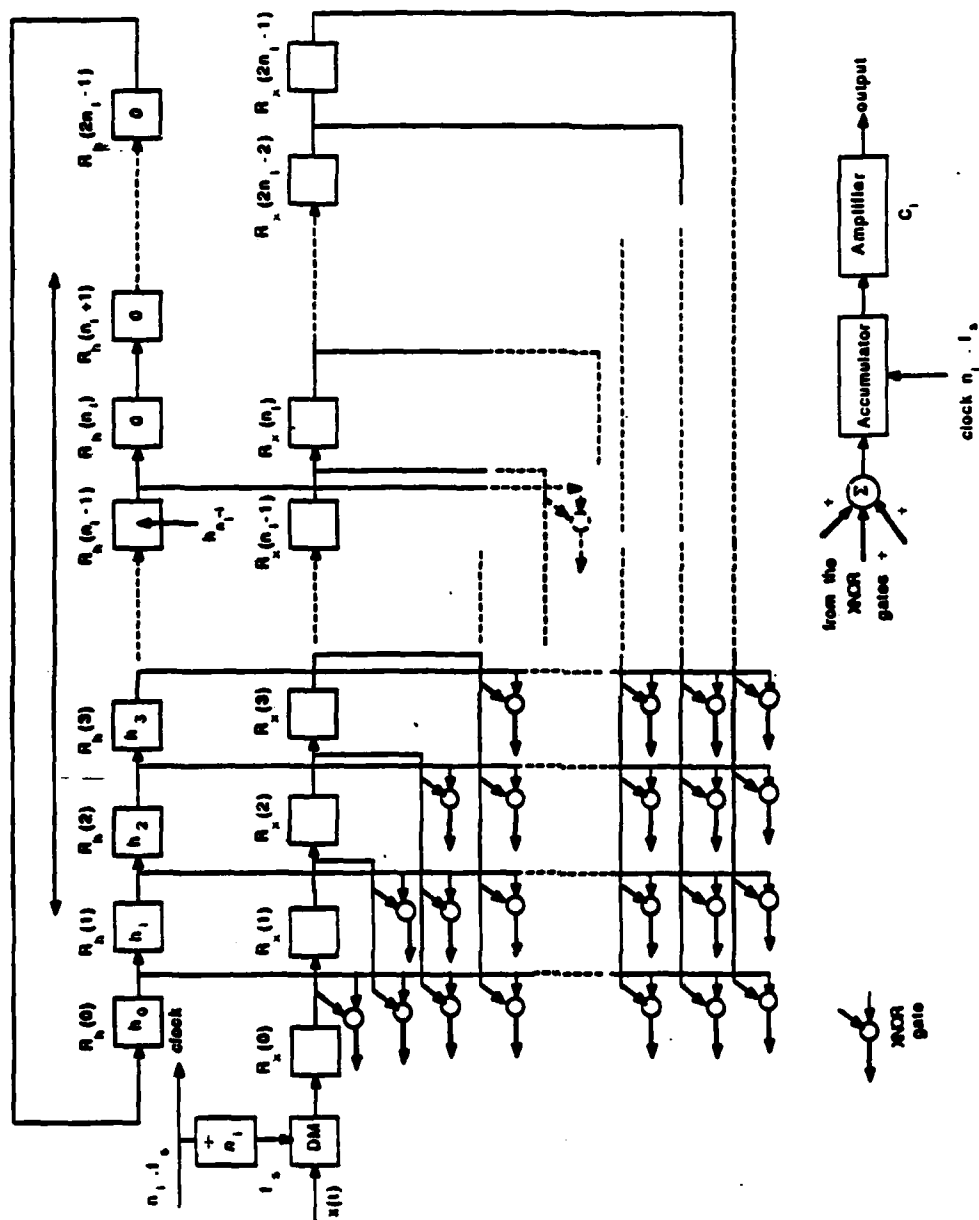
Fig. A.2. Block diagram of the direct implementation of delta convolution.

From previous assumptions, during each time period between even and odd sampling pulses the contents of R_h move to the right n_1 times producing all products needed to find the convolution, and during each time period between odd and even sampling pulses the contents of R_h move to the left n_1 times producing all products needed to find the convolution. The contents of the accumulator are multiplied by the constant C_1 to produce the convolution.

Some of the disadvantages of this direct implementation of the delta convolution are

1) By looking at the columns of the XNOR gates of Fig. A.2, we see that the needed number n_3 of binary multipliers (XNOR) gates) is

$$n_3 = n_2 + (n_2 - 1) + (n_2 - 2) + \ldots + (n_2 - n_1 + 1)$$
$$= \sum_{i=1}^{n_2} i - \sum_{i=1}^{n_2 - n_1} i$$
$$= n_1 \cdot n_2 + \frac{n_1}{2} - \frac{(n_1)^2}{2}.$$

2) The memory required is $2n_1 + n_2$ bits.

3) The high clock rate $n_1 \cdot f_s$ needed to clock R_h.

It is clear that the required number of XNOR gates and the required memory will be large, especially when n_2 is large (which is a practical assumption).

159

APPENDIX B

MODIFIED AND REDUCED DELTA CONVOLUTION METHODS

1. Modified Delta Convolution

We will derive the change in the discrete convolution $\Delta conv(i)$ when the input register R_x is fully occupied where

$$\Delta conv(i) \triangleq conv(i \cdot T_s) - conv\left([i-1] \cdot T_s\right) \qquad (B.1)$$

From Equation (3.14), at time t

$$y(t) \approx y(k \cdot T_s) = y\left([j+n_1] \cdot T_s\right) = C_1 \cdot \Bigg\{$$

$$h_0 \cdot [x_{j+n_1-1} + 2x_{j+n_1-2} + 3x_{j+n_1-3} + \ldots\ldots + (n_1 - 2) \cdot x_{j+2} + (n_1 - 1)$$

$$\cdot x_{j+1} + n_1 \cdot \sum_{i=0}^{j} x_i]$$

$$+ h_1 \cdot [\qquad x_{j+n_1-2} + 2x_{j+n_1-3} + \ldots + (n_1 - 3) \cdot x_{j+2} + (n_1 - 2)$$

$$\cdot x_{j+1} + (n_1 - 1) \sum_{i=0}^{j} x_i]$$

$$+ h_2 \cdot [\qquad x_{j+n_1-3} + \ldots + (n_1 - 4) \cdot x_{j+2} + (n_1 - 3)$$

$$\cdot x_{j+1} + (n_1 - 2) \sum_{i=0}^{j} x_i]$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ h_{n_1-1} \cdot [\qquad\qquad\qquad\qquad\qquad \sum_{i=0}^{j} x_i] \Bigg\}$$

where

$$k = \text{ integer value of } \left[\frac{t}{T_s}\right] = j + n_1$$

160

and where the needed products can be demonstrated as given in Fig. B.1. Collecting terms gives

$$y(k \cdot T_s) = y([j + n_1] \cdot T_s) = C_1 \cdot \Bigg\{$$

$$\left(h_0 \cdot \left[\sum_{i=j}^{j+n_1-1} x_i \cdot (n_1 - i + j) \right] + h_0 \cdot n_1 \cdot \sum_{i=0}^{j-1} x_i \right)$$

$$+ \left(h_1 \cdot \left[\sum_{i=j}^{j+n_1-2} x_i \cdot (n_1 - i + j - 1) \right] + h_1 \cdot (n_1 - 1) \cdot \sum_{i=0}^{j-1} x_i \right)$$

$$+ \left(h_2 \cdot \left[\sum_{i=j}^{j+n_1-3} x_i \cdot (n_1 - i + j - 2) \right] + h_2 (n_1 - 2) \cdot \sum_{i=0}^{j-1} x_i \right)$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ h_{n_1-1} \cdot \left[\sum_{i=j}^{j} x_i \cdot (n_1 - i + j - (n_1 - 1)) + h_{n_1-1} \cdot \sum_{i=0}^{j-1} x_i \right] \Bigg\}.$$

Or

$$y(k \cdot T_s) = C_1 \cdot \Bigg\{ \sum_{m=0}^{n_1-1} \left(h_m \left[\sum_{i=j}^{j+n_1-m-1} x_i \cdot (n_1 - i + j - m) \right] \right)$$

$$+ \left(\sum_{i=0}^{j-1} x_i \right) \cdot \left(\sum_{L=0}^{n_1-1} h_L \cdot [n_1 - L] \right) \Bigg\} \qquad (B.2)$$

At time $(k + 1) \cdot T_s$,

161
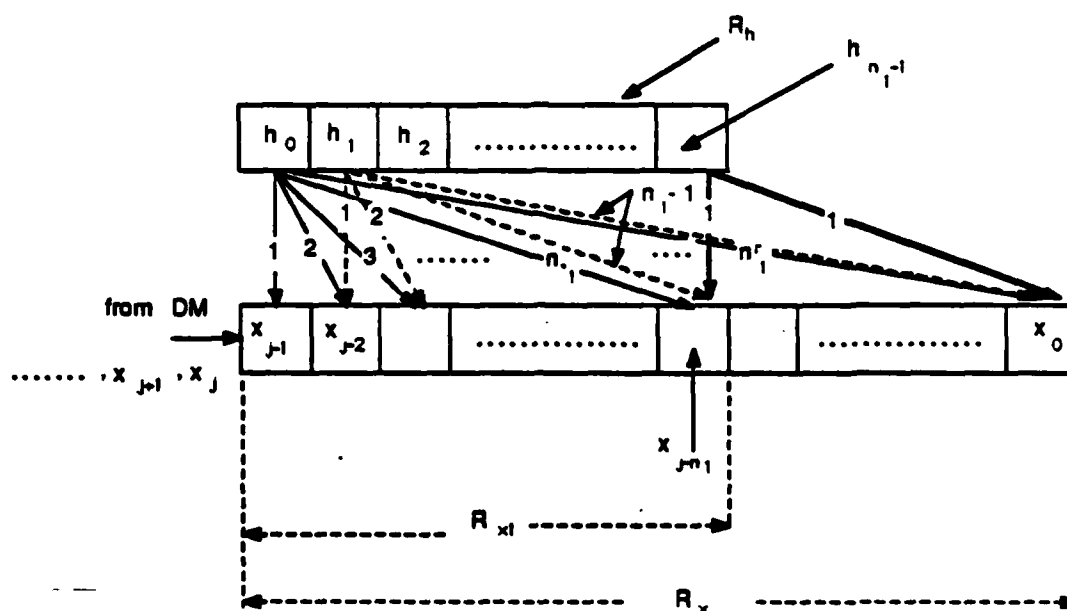
Fig. B.1. Diagram showing the needed products to find the convolution at time $(j + n_1)\,T_s$.

$$y\left([k+1]\cdot T_s\right) = y\left([n_1+j+1]\cdot T_s\right) = C_1 \cdot \Big\{$$

$$h_0 \cdot [x_{j+n_1} + 2x_{j+n_1-1} + 3x_{j+n_1-2} + \ldots\ldots + (n_1-2)\cdot x_{j+3}$$

$$+(n_1-1)\cdot x_{j+2} + n_1 \cdot \sum_{i=0}^{j+1} x_i \Big]$$

$$+ h_1 \cdot [\qquad x_{j+n_1-1} + 2x_{j+n_1-2} + \ldots + (n_1-3)\cdot x_{j+3}$$

$$+(n_1-2)\cdot x_{j+2} + (n_1-1)\cdot \sum_{i=0}^{j+1} x_i \Big]$$

$$+ h_2 \cdot [\qquad\qquad x_{j+n_1-2} + \ldots + (n_1-4)\cdot x_{j+3}$$

$$+(n_1-3)\cdot x_{j+2} + (n_1-2)\cdot \sum_{i=0}^{j+1} x_i \Big]$$

$$+\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ h_{n_1-1} \cdot \left[\qquad\qquad \sum_{i=0}^{j+1} x_i \right] \Big\}$$

$$(B.3)$$

Therefore,

$$y\left([k+1]\cdot T_s\right) = y\left([n_1+j+1]\cdot T_s\right) = C_1 \cdot \Big\{$$

$$\left(h_0 \cdot \left[\sum_{i=j}^{j+n_1} x_i \cdot (n_1-i+j+1) \right] + h_0 \left(-x_j + n_1 \cdot \sum_{i=0}^{j-1} x_i \right) \right)$$

$$+ \left(h_1 \cdot \left[\sum_{i=j}^{j+n_1-1} x_i \cdot (n_1-i+j) \right] + h_1 \left(-x_j + [n_1-1]\cdot \sum_{i=0}^{j-1} x_i \right) \right)$$

$$+ \left(h_2 \cdot \left[\sum_{i=j}^{j+n_1-2} x_i \cdot (n_1-i+j-1) \right] + h_2 \left(-x_j + [n_1-2]\cdot \sum_{i=0}^{j-1} x_i \right) \right)$$

$$+\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ \left(h_{n_1-1} \left[\sum_{i=j}^{j+1} x_i \cdot (n_1-i+j-[n_1-2]) \right] + h_{n_1-1} \left[-x_j + \sum_{i=0}^{j-1} x_i \right] \right) \Big\}.$$

Or,

$$y([k+1] \cdot T_s) = C_1 \cdot \left\{ \sum_{m=0}^{n_1-1} \left(h_m \left[\sum_{i=j}^{j+n_1-m} x_i \cdot (n_1 - i + j - m + 1) \right] \right) \right.$$

$$\left. + \left(\sum_{i=0}^{j-1} x_i \right) \cdot \left(\sum_{L=0}^{n_1-1} h_L \cdot [n_1 - L] \right) - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right\}. \qquad (B.4)$$

From Equations (B.2) and (B.4), now form $\Delta conv(j + n_1 + 1)$ where

$$\Delta conv(j + n_1 + 1) = conv([j + n_1 + 1] \cdot T_s) - conv([j + n_1] \cdot T_s)$$

$$= C_1 \cdot \left(\left\{ \sum_{m=0}^{n_1-1} \left(h_m \left[\sum_{i=j}^{j+n_1-m} x_i \cdot (n_1 - i + j - m + 1) \right] \right) - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right\} \right.$$

$$\left. - \left\{ \sum_{m=0}^{n_1-1} \left(h_m \left[\sum_{i=j}^{j+n_1-m-1} x_i \cdot (n_1 - i + j - m) \right] \right) \right\} \right)$$

$$= C_1 \cdot \left(\left\{ \sum_{m=0}^{n_1-1} h_m \left[\sum_{i=j}^{j+n_1-m} x_i \cdot (n_1 - i + j - m + 1) \right. \right. \right.$$

$$\left. \left. \left. - \sum_{i=j}^{j+n_1-m-1} x_i \cdot (n_1 - i + j - m) \right] \right\} - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right)$$

$$= C_1 \cdot \left(\left[\sum_{m=0}^{n_1-1} h_m \left\{ \left(\sum_{i=j}^{j+n_1-m-1} x_i \cdot [(n_1 - i + j - m + 1) - (n_1 - i + j - m)] \right) \right. \right. \right.$$

$$\left. \left. \left. + x_{j+n_1-m} \right\} \right] - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right)$$

$$= C_1 \cdot \left(\left[\sum_{m=0}^{n_1-1} h_m \left(\sum_{i=j}^{j+n_1-m-1} x_i + x_{j+n_1-m} \right) \right] - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right)$$

$$= C_1 \cdot \left(\left[\sum_{m=0}^{n_1-1} h_m \left(\sum_{i=j}^{j+n_1-m} x_i \right) \right] - x_j \cdot \sum_{i=0}^{n_1-1} h_i \right)$$

$$= C_1 \cdot \left(\sum_{m=0}^{n_1-1} h_m \left(\sum_{i=j+1}^{j+n_1-m} x_i \right) \right). \qquad (B.5)$$

Or,

$$\Delta conv(j + n_1 + 1) = C_1 \cdot \left(\sum_{m=0}^{n_1-1} < R_h(m) > \cdot \left(\sum_{i=m}^{n_1-1} < R_x(i) > \right) \right) \qquad (B.6)$$

where $< R_k(i) >$ means the contents of cell i of register k where k can be h or x.

We can see that the change in the discrete convolution is dependent only on the last n_1 samples of the input and is independent of all other samples. Also we can see that all the weights equal unity. The needed binary products to find the change in the discrete convolution are represented in Fig. B.2.

2. Implementation of a Transversal Filter Using the Modified Delta Convolution Method

To implement a transversal filter using Modified Delta Convolution, two methods can be used.

a. The First Method

In the first method (Fig. B.3), the needed binary multipliers (XNOR gates) $= n_1 + (n_1 - 1) + (n_1 - 2) + \ldots \ldots + 1 = \frac{n_1(2+(n_1-1))}{2} = \frac{n_1(n_1+1)}{2}$ and the needed binary storage $= 2n_1$.

b. The Second Method

In the second method (Fig. B.4) we will use the fact that we need to perform n_1 binary multiplications only and to store all the needed binary multiplications which were performed in the preceding steps. In this case the needed number of binary multipliers $= n_1$, and the needed storage is as follows:
$2n_1 + (n_1 - 1) + (n_1 - 2) + \ldots \ldots + 1 = n_1 + \frac{n_1(n_1+1)}{2} = \frac{n_1(n_1+3)}{2}$. Despite the fact that in the second method the needed hardware is reduced, we still have to use a number of XNOR gates or a number of binary storage locations proportional to $(n_1)^2$. Because of this we introduce the Reduced Delta Convolution method.

165
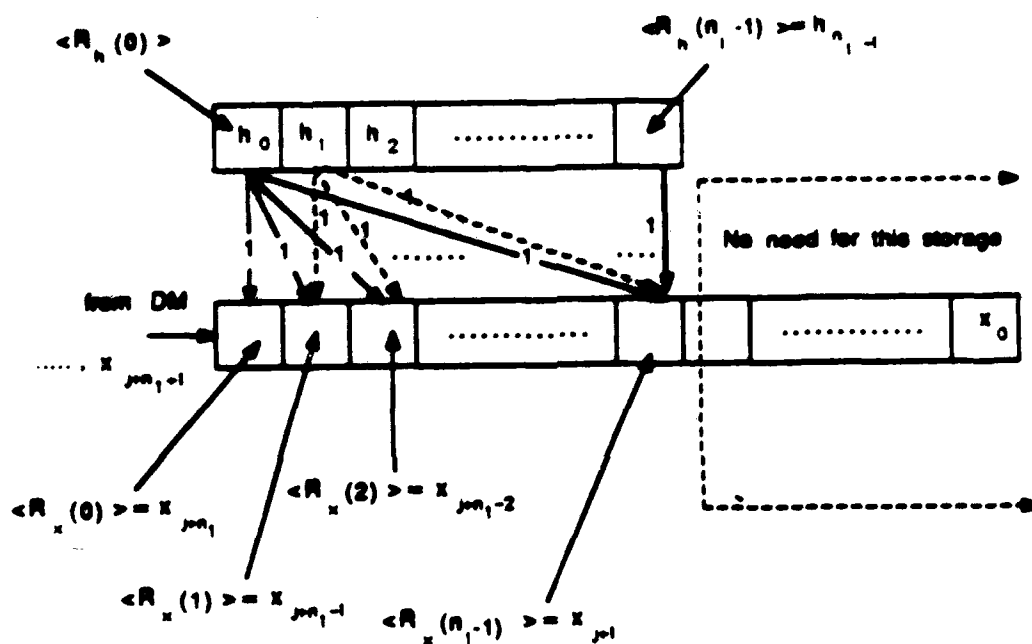
Fig. B.2. Diagram showing the needed products to find the change in the discrete convolution.

166
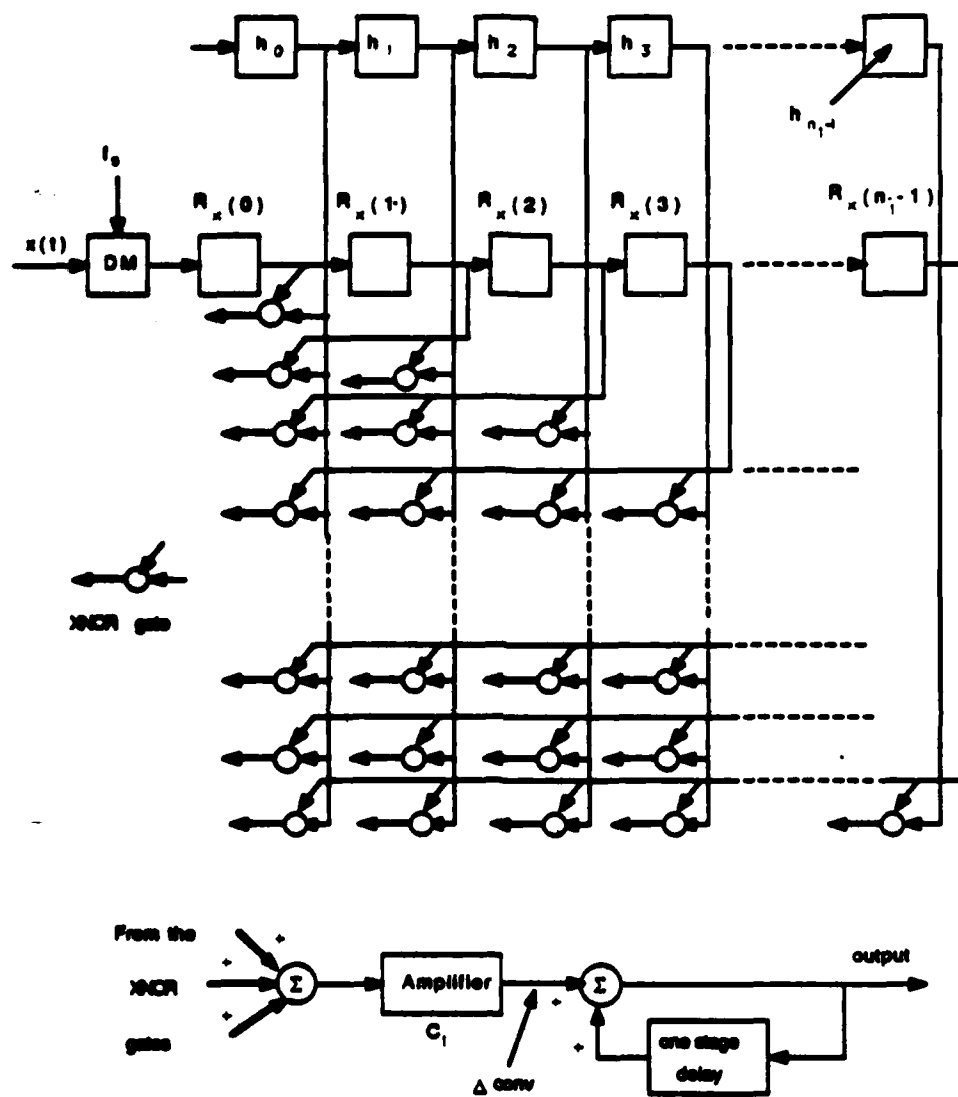
Fig. B.3. Diagram showing the first method of implementing modified delta convolution.
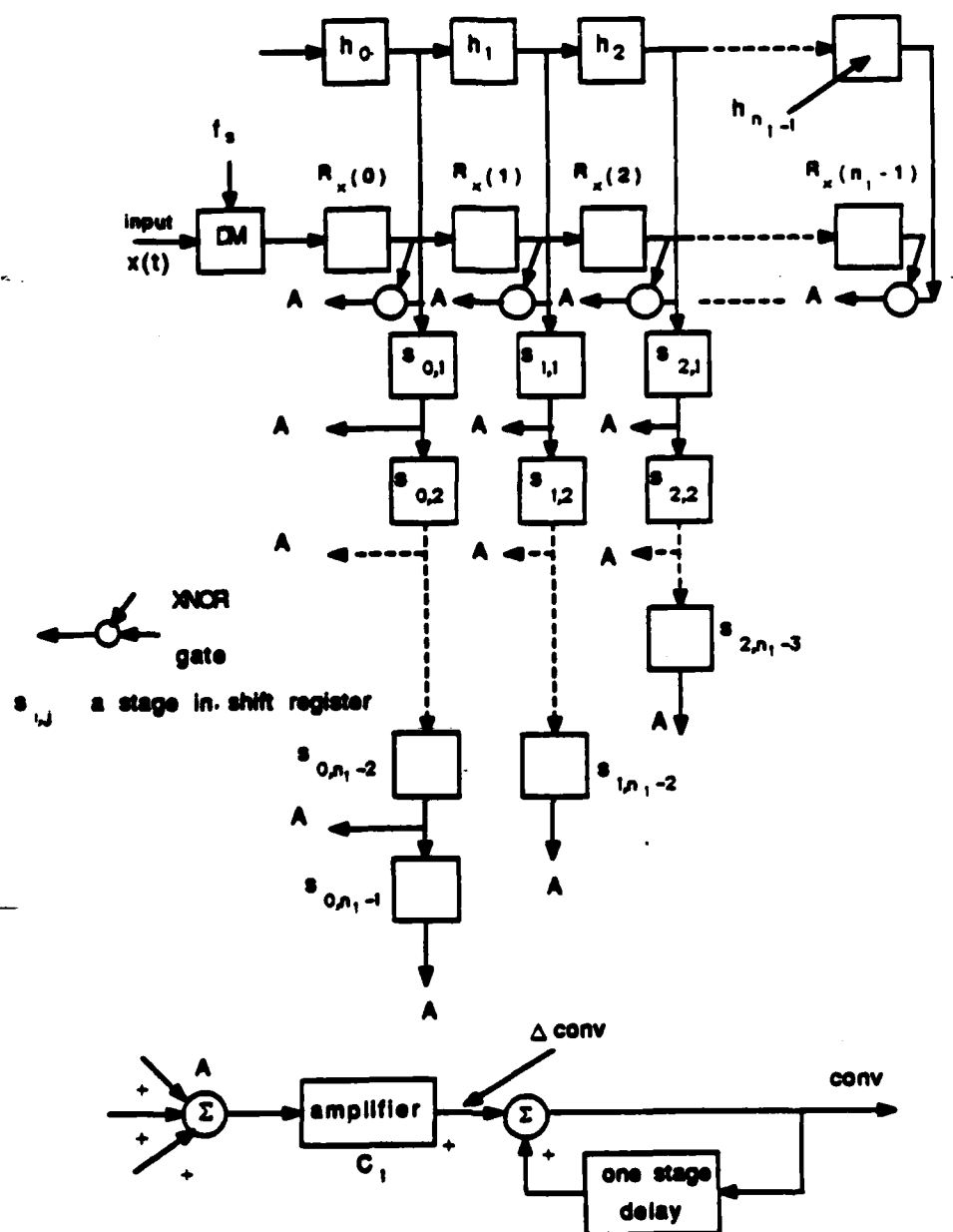
Fig. B.4. Diagram showing the second method of implementing
modified delta convolution.

168

3. The Reduced Delta Convolution (RDC) system

In this section we introduce the reduced delta convolution (RDC) method where the needed number of binary multipliers equals the filter length n_1 and the needed binary storage equals $2n_1 + 1$.

Let us define $\Delta 2conv(k \cdot T_s)$ as follows

$$\Delta 2conv(k \cdot T_s) \triangleq \Delta conv(k \cdot T_s) - \Delta conv([k-1] \cdot T_s) \qquad (B.7)$$

where $\Delta conv(k \cdot T_s)$ is the change in the discrete convolution which is defined in Equation (B.1).

We now find $\Delta 2conv(k \cdot T_s)$ when the input register is fully occupied. At time t where

$$k = \text{integer value of } \left[\frac{t}{T_s}\right] = j + n_1$$

we know that

$$\Delta conv(k \cdot T_s) = \Delta conv([j + n_1] \cdot T_s) = C_1 \cdot \Bigg\{$$

$$h_0 \left[x_{j+n_1-1} + x_{j+n_1-2} + x_{j+n_1-3} + \ldots\ldots + x_{j-1} + x_j\right]$$

$$+ h_1 \left[\qquad\quad x_{j+n_1-2} + x_{j+n_1-3} + \ldots\ldots + x_{j-1} + x_j\right]$$

$$+ h_2 \left[\qquad\qquad\qquad x_{j+n_1-3} + \ldots\ldots + x_{j-1} + x_j\right]$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ h_{n_1-1} \left[\qquad\qquad\qquad\qquad\qquad\qquad\qquad x_j\right] \Bigg\} \qquad (B.8)$$

169

and

$$\Delta conv([k+1] \cdot T_s) = \Delta conv([j+n_1+1] \cdot T_s) = C_1 \cdot \Bigg\{$$

$$h_0 \left[x_{j+n_1} + x_{j+n_1-1} + x_{j+n_1-2} + \ldots\ldots + x_j + x_{j+1} \right]$$

$$+ h_1 \left[\quad\quad\quad x_{j+n_1-1} + x_{j+n_1-2} + \ldots\ldots + x_j + x_{j+1} \right]$$

$$+ h_2 \left[\quad\quad\quad\quad\quad\quad x_{j+n_1-2} + \ldots\ldots + x_j + x_{j+1} \right]$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$+ h_{n_1-1} \left[\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad x_{j+1} \right] \Bigg\} \qquad (B.9)$$

From Equations (B.7), (B.8), and (B.9) we have

$$\Delta 2conv([k+1] \cdot T_s) = \Delta conv([k+1] \cdot T_s) - \Delta conv(k \cdot T_s)$$

$$\Delta 2conv([j+n_1+1] \cdot T_s) = \Delta conv([j+n_1+1] \cdot T_s) - \Delta conv([j+n_1] \cdot T_s)$$

$$= C_1 \cdot \Bigg\{ h_0 x_{j+n_1} + h_1 x_{j+n_1-1} + h_2 x_{j+n_1-2} + \cdots$$

$$+ h_{n_1-1} x_{j+1} - x_j \cdot \left[\sum_{i=0}^{n_1-1} h_i \right] \Bigg\} \qquad (B.10)$$

which can be represented by Fig. B.5. Referring to Fig. B.5 and Equation (B.10), then

$$\Delta 2conv([j+n_1+1] \cdot T_s) = C_1 \cdot \Bigg\{ \sum_{i=0}^{n_1-1} <R_h(i)> \cdot <R_s(i)>$$

$$- <R_s(n_1)> \cdot \sum_{i=0}^{n_1-1} h_i \Bigg\} \qquad (B.11)$$

From Equation (B.1),

$$\Delta conv([j+n_1+1] \cdot T_s) = conv([j+n_1] \cdot T_s) + \Delta conv([j+n_1+1] \cdot T_s) \quad (B.12)$$

170

and from Equation (B.7),

$$\Delta conv([j + n_1 + 1] \cdot T_s) = \Delta conv([j + n_1] \cdot T_s) + \Delta 2conv([j + n_1 + 1] \cdot T_s)$$

$$(B.13)$$

where $\Delta 2conv([j + n_1 + 1] \cdot T_s)$ is given by Equation (B.11) and where at time $t = 0$, $conv(0) = 0$.

Equations (B.11), (B.12), and (B.13) define what we call the reduced-delta convolution (RDC) system. See Fig. 4.2.
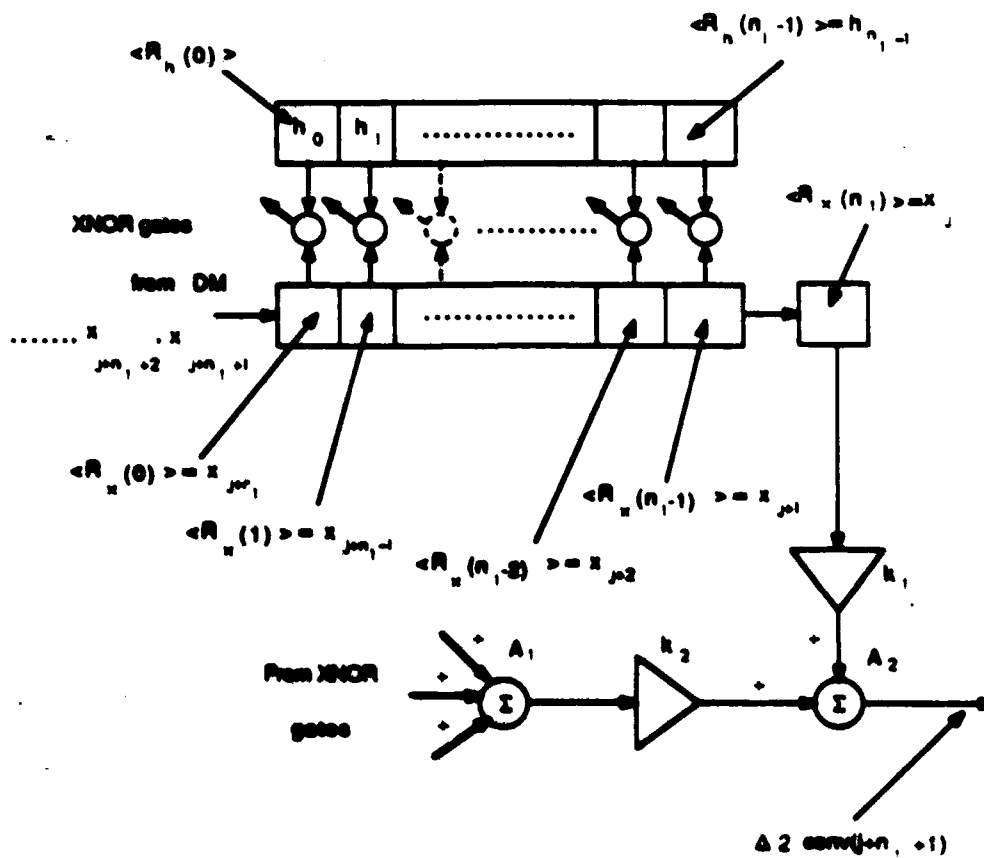
Fig. B.5. Diagram showing the creation of $\Delta 2conv\,([j + n_1 + 1] \cdot T_s)$.

172

APPENDIX C

A CONVENTIONAL MATCHED FILTER

Fig. C.1 shows a block diagram of a matched filter where

$$y(t) = \int_0^{T_d} h(\tau) \cdot w(t - \tau) d\tau$$

$$= \int_0^{T_d} x(T_d - \tau) \cdot w(t - \tau) d\tau \qquad (C.1)$$

where $x(t)$ is the signal to which the matched filter is matched and T_d is its time duration.

The first approximation involves writing a sampled-data form of Equation (C.1). See Equation (3.15).

$$y_k \approx T_s \cdot \sum_{i=0}^{L-1} x_{L-i} \cdot w_{k-i} \qquad (C.2)$$

where

$$y_j \triangleq y(j \cdot T_s), w_j \triangleq w(j \cdot T_s), \text{ and } x_j \triangleq x(j \cdot T_s)$$

The sampling rate $= \frac{1}{T_s}$ must exceed the Nyquist rate. L is the number of samples and so T_d in Equation (C.1) equals $L \cdot T_s$.

The samples x_j and w_j are now represented by binary words having elements 1 or 0 identified by x_j^m and w_j^n where

$$x_j = \sum_{m=0}^{\infty} x_j^m \cdot 2^{-m}, \quad w_j = \sum_{n=0}^{\infty} w_j^n \cdot 2^{-n} \qquad (C.3)$$

where x_j^m and $w_j^n \in \{0, 1\}$.
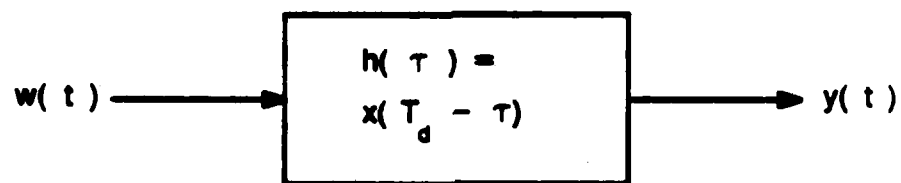
Fig. C.1. Block diagram of a matched filter.

174

We now approximate the sum in Equation (C.2) by truncating x_j to M digits and w_j to N digits. As a result of this

$$y_k \approx T_s \cdot \sum_{m=0}^{M-1} \left[\sum_{n=0}^{N-1} 2^{-(m+n)} \cdot \sum_{i=0}^{L-1} x_{L-i}^m \cdot w_{k-i}^n \right] \qquad (C.4)$$

Correlating the mth digits of the binary words of the samples of $x(\cdot)$ with the nth digits of the binary words of the samples of $w(\cdot)$ gives the digital correlator (m, n) given in Fig. C.2. At each sample time the most recent sample's n^{th} digit enters the left most stage of the nth on-line shift register. The mth digits of the sample x_j are permanently stored in the stages of the reference shift register. But, each sample w_j has M bits and each sample x_j has N bits. So, to obtain y_k we have to perform $M \cdot N$ digital correlations, where the results of correlation are weighted by the factor $T_s \cdot 2^{-(m+n)}$. We use one reference shift register, which has $M \cdot L$ binary stages, for all the $M \cdot N$ digital correlators.

Fig. C.3 shows a block diagram of the DMF for arbitrary N and L but with M equals to 1 because a block diagram for arbitrary M, N and L showing all interconnections is cumbersome. In Fig. C.3, for convenience, the gain factor T_s called for in Equation (5.8) has been left out of the weights $2^{-(m+n)}$, so the output in fact is $\frac{y_k}{T_s}$.

the nth on_line shift register

reference shift register
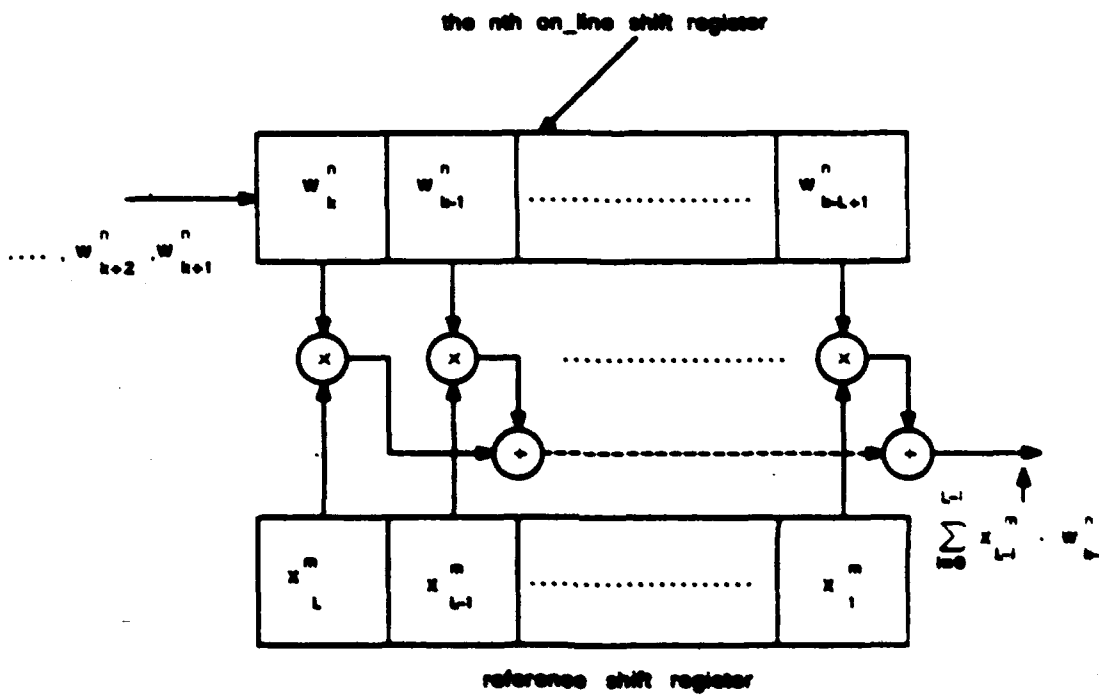
Fig. C.2. Digital correlator (m, n).
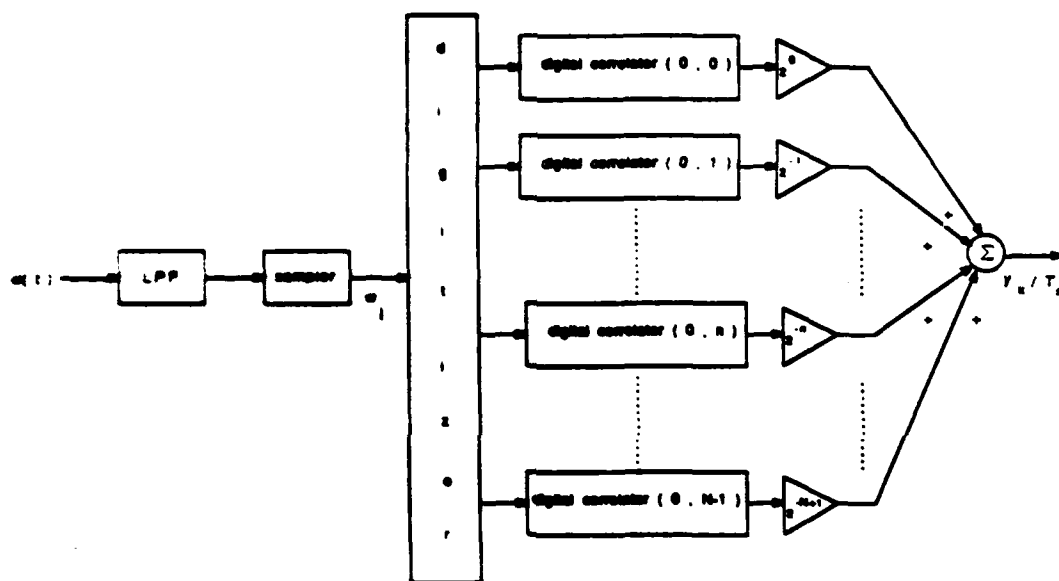
Fig. C.3. **Block diagram of a DMF** for arbitrary N and L, where $M = 1$.

APPENDIX D

DC OFFSET LEVEL

After the input expires, the zero voltage input to the delta modulator will be represented by a successive positive and negative pulses about zero. From Equation (3.18) the output of the convolver equals

$$y(t) \approx y(j \cdot T_s) = \Delta x \cdot \Delta h \cdot T_s \cdot \sum_{k=0}^{n_1-1} \left\{ \left[\sum_{p=0}^{k} h_p \right] \cdot \left[\sum_{i=0}^{j-k-1} x_i \right] \right\}; \quad j = 1, 2, 3, \ldots$$

But the input signal delta modulation pulses x_i have successive positive and negative values. Therefore,

$$\sum_{i=0}^{j-k-1} x_i = \begin{cases} 1 \text{ or } -1 & \text{for} \quad j-k = \text{odd} \\ 0 & \text{for} \quad j-k = \text{even} \end{cases} \qquad (D.1)$$

From Equation 3.11

$$h(\tau) \approx h_s(\tau) = \Delta h \cdot \sum_{p=0}^{k} h_p; \quad k \cdot T_s < \tau < (k+1) \cdot T_s \qquad (D.2)$$

Therefore, the output level at a given sample time $j \cdot T_s$ when j is odd (k even) becomes

$$T_s \cdot \left(\sum_{\substack{k=0 \\ k \text{ even}}}^{n_1-1} h(k \cdot T_s) \right) \cdot (\pm \Delta x)$$

At T_s seconds later, j will be even (k odd) and so the dc offset value is

$$T_s \cdot \left(\sum_{\substack{k=0 \\ k \text{ odd}}}^{n_1-1} h(k \cdot T_s) \right) \cdot (\pm \Delta x)$$

178

Because T_s is small when we use DM, then we can expect that the distortion will be constant and approximately equal to

$$\pm \frac{\Delta x}{2} \ (\text{total area under } h(t)) \qquad (D.3)$$

since $T_s \cdot \sum_{\substack{k=0 \\ k \text{ odd or even}}}^{\frac{n}{2}-1} h(k \cdot T_s)$ is approximately half the area of $h(t)$.

LIST OF REFERENCES

1. Kirk, D. E. and Strum, R. D., *First Principles of Discrete Systems and Digital Signal Processing*, Addison-Wesley, 1987.

2. Gordan, J. A. Bird, *Design of Continuous and Digital Electronic Systems*, McGraw-Hill, 1980.

3. Rabiner, L. R. and Gold, B., *Theory and Application of Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, N. J., 1975.

4. Kailath, Thomas, *Modern Signal Processing*, Hemisphere Publishing Corporation, 1985.

5. Hamming, R. W., *Digital Filters*, Prentice-Hall, 1977.

6. Young, T., *Linear Systems and Digital Signal Processing*, Prentice-Hall, 1985.

7. McClellan, James H. and Parks, Thomas W., "A Unified Approach to the Design of Optimum FIR Linear-phase Digital Filters," *IEEE Trans. Circuit Theory*, CT-20 (6), pp. 697–701, 1973.

8. Whalen, H. H., *VLSI Applications and Testing in Very Large Scale Integration*, D. F., Ed., Springer-Verlag, 1985.

9. Turin, G. L., "An Introduction to Matched Filters," *IRE Trans. Inform. Theory*, vol. IT-6, pp. 311–329, June, 1960.

10. Turin, G. L., "An Introduction to Digital Matched Filters," *Proceedings of IEEE*, vol. 64, No. 7, July 1976.

11. Horrigan, Frank A., "Signal Processing Fundamentals," *Electronic Progress*, vol. XXVIII, No. 1, pp. 9–10, 1987.

12. Boyle, W. S. and Smith, G. E., "Charge Coupled Semiconductor Devices," *Bell Syst. Tech. J.*, vol. 49, pp. 587–593, April 1970.

13. Amelio, G. F., Tompsett, M. F., and Smith, G. E., "Experimental Verification of the Charge Coupled Device Concept," *Bell Syst. Tech. J.*, vol. 49, pp. 593–600, April 1970.

14. Tomsett, M. F., "Charge Transfer Devices," *J. Vac. Sci. Technol.*, vol. 9, pp. 1166–1181, July/August 1972.

15. Sangster, F. L. J., "Integrated MOS and Bipolar Analog Delay Lines Using Bucket-Brigade Capacitor Storage," IEEE Solid-State Circuits Conf., *Dig. Tech. Papers*, pp. 74–75, 1970.

16. Boonstra, L. and Sangster, F. L. J., "Progress on Bucket-Brigade Charge-Transfer Devices," IEEE Solid-State Circuits Conf., *Dig. Tech. Papers*, pp. 140–141, 1972.

17. Butler, W. J., Puckette, C. M., Barron, M. B., and Kurtz, B., "Analog Operating Characteristics of Bucket-Brigade Delay Lines, "IEEE Solid-State Circuits Conf., *Dig. Tech. Papers*, pp. 138–139 and 226–227, 1972.

18. Buss, D. D., Collins, D. R., Bailey, W. H., and Reeves, C. R., "Transversal Filtering Using Charge-transfer Devices," *IEEE J. Solid-State Circuits*, vol. SC-8, No. 2, April 1973.

19. Bosshart, P., "An Integrated Analog Correlator Using Charge Coupled Devices," in *Proc. IEEE Int. Solid-State Circuits Conf.*, pp. 198–199, 1976.

20. Denyer, P. B., Mavor, J., and Arther, J. W., "Miniature Programmable Transversal Filter Using CCD/MOS Technology, " *Proc. IEEE*, vol. 67, pp. 42–50, January 1979.

21. Tiemann, J. J., Engeler, W. E., and Baertsch, R. D., "A Surface-charge Correlator," *IEEE J. Solid-State Circuits*, vol. SC-9, pp. 403–409, December 1974.

22. Tanaka, S. C., Tseng, H. F., Lin, L. T., and Chen, P. T., "A Real-time programmable Transversal Filter," *IEEE J. Solid-State Circuits*, vol. SC-15, No. 6, December 1980.

23. Gandolfo, D. A., Tower, J. R., Pridgen, J. I., and Munroe, S. C., "Analog-binary CCD Correlator: A VLSI Signal processor," *IEEE J. Solid-State Circuits*, vol. SC-14, pp. 518–525, April 1979.

24. Creasey, D. J., *Advanced Signal Processing*, Short Run press Ltd., London, pp. 85–86, 1985.

25. Haken, R. A., Pettengill, R. C., and Hite, L. R., "A General Purpose 1024-Stage Electronically Programmable Transversal Filter," *IEEE J. Solid-State Circuits*, vol. SC-15, No. 6, December 1980.

181

26. Terrell, T. J., *Introduction to Digital Filters*, John Wiley, Inc., 1980.

27. Roden, Martin S., *Digital and Data Communication Systems*, Englewood Cliffs, N. J.: Prentice-Hall, 1982.

28. Ha, Tri T., *Digital Satellite Communications*, Macmillan, New York, 1986.

29. Goodman, D. J., "Delta Modulation Granular Quantizing Noise," *Bell Syst. Tech. J.*, vol. 48, No. 5, pp. 1197–1218, May/June 1969.

30. Greenstein, L. J., "Slope Overload Noise in Linear Delta Modulators with Gaussian Inputs," *Bell Syst. Tech. J.*, vol. 52, No. 3, pp. 387–421, March 1973.

31. Jayant, N. S., *Digital Coding of Waveforms, Principles and Applications to Speech and Video*, Prentice-Hall, 1984.

32. Peebles, P. Z., *Digital Communication Systems*, Prentice-Hall, 1987.

33. Protonotarios, E. N., "Slope Overload Noise in Differential Pulse Code Modulation Systems," *Bell Syst. Tech. J.*, vol. 64, No. 9, pp. 2119-2161, November 1967.

34. Zetterberg, L. H., "A Comparison Between Delta and Pulse Code Modulation," *Ericsson Technics*, No. 1, pp. 95–154, January 1955.

35. O'Neal, J. B., Jr., "Delta Modulation quantizing Noise analytical and Computer Simulation Results for Gaussian and Television Input signals," *B.S.T.J.*, 45, No. 1, pp. 117–141, January 1966.

36. Abate, J. E., "Linear and Adaptive Delta Modulation," *Proc. IEEE*, 55, pp. 298–308, March 1967.

37. Denyer, Peter, *VLSI Signal Processing: A bit serial approach*, Addison-Wesley, 1985.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Ahmed Fahmy Amin 102 Madinat El-Tawfik El-Tairan Street Apartment #22 Cairo, Egypt	6
4.	Minister of Defense Cairo, Egypt	1
5.	Egyptian Military Attaché Egyptian Military Office 2308 Tracy Place, N. W. Washington, D. C. 20008	1
6.	Commandant of Military Technical College Cairo, Egypt	1
7.	Prof. Mohammed Fahmy Amin Military Technical College Cairo, Egypt	1
8.	Dr. Mostafa Fahmy A. 9A, Mahmoud Sidky Street OMRANIA Giza, Egypt	2
9.	Mohammed K. Abd-ElMoniem 27, Ahmed Maher Street Giza, Egypt	2

10. Prof. Farid M. Badran   2
    Electrical Engineering Department
    Cairo University
    Gisa, Egypt

11. Prof. John P. Powers   1
    Chairman, Electrical and
    Computer Engineering
    Code 62Po
    Naval Postgraduate School
    Monterey, California 93943-5000

12. Prof. G. A. Myers   5
    Code 62Mv
    Naval Postgraduate School
    Monterey, California 93943-5000

13. Prof. D. E. Kirk   1
    Code 62Ki
    Naval Postgraduate School
    Monterey, California 93943-5000

14. Prof. R. W. Hamming   1
    Code 52Hg
    Naval Postgraduate School
    Monterey, California 93943-5000

15. Prof. F. W. Terman   1
    Code 62Tz
    Naval Postgraduate School
    Monterey, California 93943-5000

16. Prof. H. M. Fredricksen   1
    Code 62Fs
    Naval Postgraduate School
    Monterey, California 93943-5000

17. Commander   1
    Space and Naval Warfare
    Systems Command
    Washington, D.C. 20363-5100
    Attn: Capt Stacy V. Holmes

END

FILMED

FEB. 1988

DTIC